

COMPUTER PROGRAM LISTING APPENDIX

Application Overview

e-LMO Fast Track is a web based application which will be delivered to approved Servicers via the GE MI website Servicers' Page. The intent of this web application is to provide servicers with a web-based mechanism for handling Reactive Workout Requests.

A user in a servicer shop will be able to initiate a Reactive Workout AND receive an Instant Approval via the web. The user will enter a certificate number and choose the desired Workout Type. The application will retrieve origination and delinquency data from our mainframe datastores and present this info to the user. The user will use this information to a) verify that this is the correct certificate for the workout and b) update the displayed information with the more up-to-date servicer data (if needed). The user will be required to enter a few pieces of data (data required varies based on workout type selected). This workout specific data along with the other delinquency and origination data will be run through e-LMO Fast Tracks minimum guidelines (by workout type) to determine if the workout request should be approved. If approved, the user will be shown the conditions of approval. If the user accepts the approval, they can then print the approval or save it as a file. Once the user accepts the approval, REO will be updated to indicate that the workout has been started by a Fast Track user. The application will also interface with WorkFlow Manager to initiate followup tasks corresponding to the conditions on the approval letter. If the workout request does not meet the minimum guideline edits, the user will be shown a screen indicating such with a short explanation. The user will have the option to , (based on authority), proceed with full data entry (i.e. regular LMO) or to send the package in. If they opt for full data entry, they will be presented (at least currently) with the LMO login screen. If they opt to send the package in, a screen will show them which documents they should fax/mail.

Additional support for the user will be on the Contact Us tab which will give them the phone numbers, email links direct to workout reps here, mailing address, fax info, etc. Each user will also be able to view a list of all workouts that they put into Fast Track or all Fast Track workouts for a specific certificate number. By selecting an entry from the list presented, the user will be able to view the data entered into Fast Track for that certificate. All documents that are generated from the application, such as Promissory Note, Approval Letters and checksheets , will be able to be printed/saved by the user as well as be re-generated at any later time either by that user or one here at the National Loan Workout Center (NLWC).

For a specified period of time, users will be receive e-wards incentives for using the application. A link will be provided on the e-LMO Fast Track Main Page to allow users to get the e-wards website. Here they will be able to register for the program, view program rules, view their e-points balance and order e-ward items.

Reporting capabilities from the application will include ?????

User accounts will be setup and assigned by the Action Center. When the user goes through the MI Servicer Logon, they will then have access to information and applications on the Servicer page based on their authority level. Users who are authorized for e-LMO Fast Track will be able to click on it here to start the process.

General Form Specifications

1. all fonts should be Arial
2. drop boxes should show contents of entire list – no spin controls, no mouse overs.
3. data entry fields should be readily distinguished from REO/RMG data, by color coding

MAIN PAGE

Form Layout

see p:\misso\workout\elmo DMADV\Fast Track Demo\mainpage.htm

Form Components

5 buttons , see form links below

see screenfields.xls workbook, MainPage worksheet MainPage for more details on the HTML template

Certificate Number – Alphanumeric text field. Data entered by user. REQUIRED.

Lender Loan # - Alphanumeric text field. Data entered by user. OPTIONAL

Investor Name – single select listbox. Prefill box with 'Fannie Mae' , 'Freddie Mac' , 'Other' in that order

Workout Type – Single Select list box. User must choose ONE from the list. Show all options in list, do not allow a scroll bar.

Continue Button – see Form Process below.

Form Links

NOTE - buttons should be for links to

Contact Us

Submit Package

Fast Track Workouts

e-Wards

Logoff

To return to Servicer Site Home page, click on Logoff button.

If you do not see a workout listed that you would like approval on, click on the Contact Us button to contact GE for assistance.

If you would like to submit a package for a GE Workout Rep to review, click on the Submit Package button for details on package submission.

Form Processes

Read Registry Entries

path for HTML Templates

eLMOs userid and password to login to NOVA

FTRK_Interface_Ver (should be set to FTRK 1.0.0 on MTS box)

FTRK_Data_Layer_Ver (should be set to FTRK 1.0.0 on MTS box)

rest to be determined

Load Form

Store User_id and OrgNbr (passed as parameters in the URL from signon asp) from the Servicer webpage at start of e-lmoFastTrack as

applSecUserId as String

applSecOrgId as String

Using UserUtility .dll

Get phoneNbr for applSecUserID and save as VB Variable

parse URL parameters and store userid and org id in above variables (these variable names will be referenced throughout the various spec documents).

FTRK 1.0.0

```

' check for valid CertNumber
  validCert = True
  'validate Certificate Number – cert number entered should be 10 digits
  If len(certNbr) = 0 then

```

```

Error Message to User ' Please enter a 10 digit Certificate Number'
validCert = False
redisplay mainPage form
Else
    If len(certNbr) <> 10 then
        add 0 to front of certNbr
        If len(certNbr) <> 10 then
            add 0 to end of certNbr
            If len(certNbr) <> 10 then
                Error Message to user 'Please enter a 10 digit Certificate Number'
                validCert = False
                redisplay mainPage form
            End If
        End If
    End If
End If

' ensure User chooses a workout type
If a requestedWO was not selected from droplist then
    Error Message to user 'Please choose a Workout Type.'
    redisplay mainPage form
End If

'check for other FastTrack workouts for this certificate in the LMO loans table
' format SQL Query to get the Loan Data. If wo_approved_ind = 'P' then continue
' to get borrower and subject property and requested workout data to display all elements of
' Orig And Delq Data and workout data on screen.
If validCert then
    'Check LMO to see if there is a Workout In progress in the Oracle DB
    GetLoanData
    If row returned then
        '*** possibly add check here for (current date - input date < some # days to prevent '
        'workout to be approved on old data.
        'then workout information was inserted into LMO but has not been approved
        'user can continue working on this workout
        getBorrowerData
        getPropertyData
        Select Case loans.workout_cd
            Case 'MOD'
                getModData
                load MOD screen as new window
            Case 'Repay/Forbear'
                getRepayData
                load Repay screen as new window
            Case 'BAP'
                getBAPdata
                load BAP as new window
            Case 'BAPMOD'
                getBAPMOD data
                load BAPMOD as new window
            Case 'Presale'
                getPresaleData
                load Presale as new window
            Case 'Deed-in-lieu'
                getDILData

```

load DIL as new window

End Select

Else

'LMO does not have any workouts in progress for this cert. Need to go get data from
'REO

Call CheckInvestorName

Call GetOrigAndDelqData

If errors from above go to Error_Exit

'Check to see if REO shows an approved/not completed workout for this
'certificate

If investorName selected from dropbox <> investor_name returned from REO
then

'reset the investor loan number returned from REO to 'blank'

investor_loan_nbr returned from REO = ' '

'call REO Update for Investor Name Change

call REOUpdate('INVCHANGE', cert_nbr, master_policy_nbr,
REOInvestorName, workoutPgmInd, investor_loan_nbr, 'F-
EFT', systemDate)

Insert into message_info values (@sysDateTime, @siteID, @certNbr,
'REOUPDT', @errorCode, '@dataField ', '@valueOfDataField ')
as returned in error record from REOUpdate pgm. SAME error record
as Extract.

End If

If (workout_status_ind from REO = 'P') then

alertMsg 'An Approved ' & REO data workout_cd & ' Workout
already exists for this certificate. Would you like to fail this one and
enter a new workout?' with Yes or No buttons (or OK or Cancel)

If NO then

keep user at the MainPage - blank out Certificate Nbr and
Lender Loan Number and reset Type of Workout to its default

End If

If YES then

'call REO Update transaction to show the Workout in REO as failed
Call REOUpdate(certNbr, masterPolicyNbr, 'F' for status,
by EFT, date system date)

'prepare to insert a new workout into LMO

Call GetNextID

Call GetSeverityForCert

Call InsertLoan

Call InsertBorrowers

Call InsertProperty

'this function will check mainframe/user entered data to

'Requested Workout type for feasibility -all the

'feasibility edits for a specific WO type are contained in

'this function

Call Function to ValidateWORequest

If Error returned then

go to Error_Exit

Else

prepare the Header Screen with the REO data returned

End If

Select Case requestedWO

05/03/2010 02:16:01

```

        Case 'MOD'
            load MOD screen as new window
        Case 'Repay/Forbear'
            load Repay screen as new window
        Case 'BAP'
            load BAP as new window
        Case 'BAPMOD'
            load BAPMOD as new window
        Case 'Presale'
            load Presale as new window
        Case 'Deed-in-lieu'
            load DIL as new window
        End Select
    end if
Else
    'no pending workouts exist in LMO and no approved and not closed workouts
    'exist in REO
    'prepare to insert a new workout into LMO
    Call GetNextID
    Call GetSeverityForCert
    Call InsertLoan
    Call InsertBorrowers
    Call InsertProperty
    'this function will check mainframe/user entered data to
    'Requested Workout type for feasibility --all the
    'feasibility edits for a specific WO type are contained in
    'this function
    Call Function to ValidateWORequest
    If Error returned then
        go to Error_Exit
    Else
        prepare the Header Screen with the REO data returned
    End If
    Select Case requestedWO
    Case 'MOD'
        load MOD screen as new window
    Case 'Repay/Forbear'
        load Repay screen as new window
    Case 'BAP'
        load BAP as new window
    Case 'BAPMOD'
        load BAPMOD as new window
    Case 'Presale'
        load Presale as new window
    Case 'Deed-in-lieu'
        load DIL as new window
    End Select
End If
End If
End If

Error_Exit:
    Handle Error as follows:
    SQL Insert: For each cert which errors, insert to the Oracle LMO
    database table message_info. See ErrorRecord definition.

```

10/27/2010 10:24:01

Insert into message_info values (@sysDateTime, @siteID, @certNbr, 'EXTRACT',
@errorCode, '@dataField ', '@valueOfDataField ')

(see error messages in the err_codes table for description corresponding to code. For
more information of what error codes are raised see specsREOExtract.doc)

select err_message_txt from err_codes where err_code_txt = @errorCode
Build DoesNotMeetGuidelines screen (see specsDoesNotMeet.doc for more information)

serrCode = workout.err_codes.err_code_txt
serrMsg = workout.err_codes.err_message_txt

Functions

Function CheckInvestorName

' this function will check to see if the investor selected is part of a delegated program
' If the user chooses Freddie Mac then, error -> not done in Fast Track, see error message
' If the user chooses FNMA, then, using the applSecOrgID assigned to the user logged in, check
' the WorkFlow Acct Mgr Table to determine is workout_pgm_code = 'A' for Active Delegated
' Partner

If investorName selected in drop box = 'Freddie Mac' then

REOInvestorName = 'FHLMC'
Insert into message_info values (@sysDateTime, @siteID, @certNbr, 'FASTTRACK',
'103', 'InvestorName ', '@selected value of InvestorName dropbox ')

select err_message_txt from err_codes where err_code_txt = '103'

Build DoesNotMeetGuidelines screen (see specsDoesNotMeet.doc for more information)

Else

If investorName selected in drop box = 'Fannie Mae' then

If IsActiveDelegatedOrg(applSecOrgID) then

REOInvestorName = 'FNMA - A'

Insert into message_info values (@sysDateTime, @siteID, @certNbr,
'FASTTRACK', '103', 'InvestorName ', '@selected value of
InvestorName dropbox ')

select err_message_txt from err_codes where err_code_txt = '103'

Build DoesNotMeetGuidelines screen (see specsDoesNotMeet.doc for
more information)

Else

REOInvestorName = 'FNMA - P'

End If

End If

End If

Function IsActiveDelegatedOrg (orgID as string)

'using WorkFlow table, check workout_pgm_ind

' this table is still being designed

DEFAULT to return FALSE for now.

Freddie Mac

```

Select workout_pgm_ind from ?????????? where org_id = orgID
If workout_pgm_ind = 'A' then
    return True
else
    return False
End If

```

Function GetLoanData

```

Select internal_id_nbr, workout_cd, user_id, input_date, current_upb_amt, coverage_pct,
est_fclsr_start_date, fclsr_sale_date from loans
Where cert_nbr = @dataEnteredCertNbr and wo_approved_ind = 'P'

If substr(oLoans.workout_cd,1,3) = 'MOD' then
    requestedWO = 'MOD'
Else
    If substr(oLoans.workout_cd, 1,3) = 'BAP' then
        If substr(oLoans.workout_cd,1,6) = 'BAPMOD' then
            requestedWO = 'BAPMOD'
        Else
            requestedWO = 'BAP'
        endif
    Else
        requestedWO = oLoans.workout_cd
    end if
end if

```

Function GetNextId

```

'get the next id then immediately update the table to set the next avail ID
select NEXT_ID from id_counter where TABLE_NAME = 'LOANS' and
    COLUMN_NAME = 'INTERNAL_ID_NBR'

```

```
oLoans.internal_id_nbr = id_counter
```

```

update id_counter Set NEXT_ID = NEXT_ID + 1 where TABLE_NAME = 'LOANS' and
    COLUMN_NAME = 'INTERNAL_ID_NBR'

```

Function GetSeverityForCert

```

Select severity from state where state = PropertyAddress State code and
    Property Address Zip Code between begin_zip_cd and end_zip_cd

```

Function GetOrigAndDelqData

```
Format the following
```

cert_nbr	char(20)	as data entered by user
delq_org_id	char(20)	passed from login to site
lender_loan_nbr	char(20)	as data entered by user

```
Call 'black box' in GEMISSODAL
```

```
Move the temp_string returned from the blackbox call as follows
```

```

If 1st 3 characters of temp_string = 'ERR' then
    move temp_string to the errorRecord definition
Else

```


move to the loans, borrowers and property data structures (see below)

End If

'Insert into oracle

Call InsertLoan

Call InsertBorrowers

Call InsertProperty

'Extract Data String to be formatted for reading as follows

' length = 347

Private Type REO_Loan

```

cert_nbr           As String * 10
lender_loan_nbr    As String * 20
orig_upb_amt       As String * 12
orig_term_nbr      As String * 3
orig_date          As String * 10
orig_int_rate_pct  As String * 6
reinstate_amt      As String * 10
loan_due_date      As String * 10
mth_delq_nbr       As String * 3
curr_upb_amt       As String * 10
remain_term        As String * 3
curr_int_rate_pct  As String * 6
fclsr_start_date   As String * 10 /*oLoans.override_fclsr_date
fclsr_sale_date    As String * 10 /*oLoans.fclsr_sale_date
pi_amt             As String * 10
ti_amt             As String * 9
piti_amt           As String * 10
coverage_pct       As String * 3
category_cd        As String * 40
investor_name      As String * 40
previous_bap_advance_amt As String * 10
insurance_type_cd  As String * 1
investor_loan_nbr  As String * 10
user_servicer_name As String * 40
delq_servicer_org_id As String * 20
workout_cd         As String * 20
workout_status_ind As String * 1
master_policy_nbr  As String * 10

```

End Type

' length = 374

Private Type REO_Borrower

```

borr_cd           As String * 3
pri_last_name     As String * 20
pri_first_name    As String * 20
pri_middle_init   As String * 1
pri_phone_nbr_txt As String * 25
pri_chp7_bankruptcy_ind As String * 1
bankruptcy_status As String * 20
pri_borr_addr1_txt As String * 65
pri_borr_addr2_txt As String * 65
pri_borr_city     As String * 50
pri_borr_state    As String * 2
pri_borr_zip      As String * 10

```

```

cob_last_name      As String * 20
cob_first_name     As String * 20
cob_middle_init    As String * 1
oth_last_name      As String * 20
oth_first_name     As String * 20
oth_middle_init    As String * 1
bk_rlse_date       As String * 10
End Type

```

```
' length = 232
```

```

Private Type REO_Prop
    addr1_txt      As String * 65
    addr2_txt      As String * 65
    city           As String * 50
    state          As String * 2
    zip_cd         As String * 10
    occupancy_des  As String * 40
End Type

```

```

Private Type REO_Workout
    REO_Loan As REO_Loan
    REO_Borrower As REO_Borrower
    REO_Prop As REO_Prop
End Type

```

```
' Total lenght of the extract
```

```

Private Type REO_Workout_Buffer
    sOutPutBuffer As String * 953
End Type

```

```

Private Type REO_Err_record
    err_ind      As String * 3
    err_datetime As String * 18
    err_cert     As String * 10
    err_code     As String * 10
    err_data     As String * 40
    err_field    As String * 65
End Type

```

```

Private Type REO_Err_record_Buffer
    sErrBuffer As String * 146
End Type

```

```
GetFTMaxValues
```

'Note – these values will need to be accessed throughout the application for various calcs
 Select ft_Max_Repay_Term, ft_Max_BAP_Exposure_pct, ft_NOS_Pct, ft_FMV_Pct,
 ft_Total_Debt_Pct, ft_Loss_Limit_Amt, ft_Max_Nbr_BPO_Days, ft_UPB_Tolerance_Pct,
 ft_failed_wo_eward_amt, ft_submit_wo_eward_amt, ft_full_data_entry_eward_amt,
 ft_approve_wo_eward_amt, ft_noncure_note_start_days, ft_cure_note_start_days
 from max_values

```
InsertLoan (table Name workout.loans)
```

```

    internal_id_nbr = id number returned from GetInsertLoanID
    cert_nbr = cert # entered on MainPage and stored as global variable
    user_id = applSecUserID

```

```

site_id = userSiteID
workout_cd = 'SYSBLK'
user_sign_off_id = NULL
workout_sign_off_date = Null
interface_ver = FTRK_Interface_Ver from registry
data_layer_ver = FTRK_Data_Layer_Ver from registry
loan_Locked_ind = ' '
input_date = current System date
orig_upb_amt = orig upb amt returned from REO
orig_upb_des = 'System'
orig_term_nbr = orig term nbr returned from REO
int_rate_pct = returned from REO
reinstate_amt = returned from REO
loan_due_date = returned from REO
loan_due_date_des = 'System'
est_mth_delq_nbr = mth_delq_nbr returned from REO
mth_delq_nbr = same as above
curr_upb_amt = returned from REO
curr_upb_amt_des = 'System'
remain_term_nbr = orig_term_nbr - (#months difference between loan_due_date
                                and orig_date)
remain_term_nbr_des = 'System'
est_term_remain_nbr = same as remain_term_nbr
curr_int_rate_pct = 0
curr_int_rate_pct_des = 'N/A'
est_payoff_date = '01/01/1900' oracle default 'blank' date
est_payoff_amt = 0
actual_payoff_amt = 0
actual_payoff_amt_des = 'N/A'
est_fclsr_date = '01/01/1900'
override_fclsr_date = Foreclosure Start Date as returned from REO
override_fclsr_date_des = 'System'
act_pi_amt = returned from REO
est_pi_amt = same as above
pi_des = 'System'
ti_amt = returned from REO
piti_amt = returned from REO
coverage_pct = returned from REO
category_cd = returned from REO
reactive_workout_req_des = 'N/A'
investor_name = investor Name selected in dropbox from the MainPage Form
investor_allows_mods_ind = 'Y'
min_inv_int_rate_pct = 0
workout_plan_des = 'N/A'
wo_approved_ind = 'P'
wo_status_ind = 'P'
orig_ltv = 0
import_ind = 'N'
advantabe_exp_pct = 0
previous_bap_adv_amt = returned from REO
insurance_typ_cd = (select insurance_typ_cd from site_info where site_id = userSiteID)
servicer_loan_nbr = returned from REO
investor_loan_nbr = (If investor_name returned from REO = Investor Name selected on
MainPage then set investor_loan_nbr = returned from REO value Else set to spaces)
claim_deal_ind = ' '

```

fcslr_sale_date = returned as fcslr_sale_date from REO

FOR all remaining columns set as their corresponding data type default
 varchar or char = ' ' (NOTE if column name ends in _des set to 'N/A'),
 date = '01/01/1900',
 float = 0
 number = 0

Insert Borrowers(table Name workout.borrowers)

**** REPEAT FOR EACH BORROWER RETURNED FROM THE REO EXTRACT FOR A
 MAX OF 3 BORROWERS

internal_id_nbr = same as used in Insert Loan

borr_cd = returned from REO (PRI, COB, OTH depending on the number of Borrowers existing
 on the mortgage)

last_name = returned from REO

first_name = returned from REO

middle_init = returned from REO

honorific_txt = ' '

phone_nbr_txt = returned from REO for borr_cd = 'PRI' ONLY

chapter7_ind = Chp7_bankruptcy_ind returned from REO for borr_cd = 'PRI' ONLY

chapter7_des = If chapter7_ind = 'Y' then set = 'System' else set = 'N/A' for borr_cd = 'PRI'

ONLY

borr_addr_txt1 = returned from REO for borr_cd = 'PRI' ONLY

borr_addr_txt2 = returned from REO for borr_cd = 'PRI' ONLY

borr_addr_city = returned from REO for borr_cd = 'PRI' ONLY

borr_addr_state = returned from REO for borr_cd = 'PRI' ONLY

borr_addr_zip = returned from REO for borr_cd = 'PRI' ONLY

servicer_bkrupt_status_ind = 'blank' for borr_cd = 'PRI' ONLY)

other_bankruptcy_des = (If bankruptcy_status returned from REO is = 'Chapter 11' or 'Chapter
 13' then set this = to bankruptcy_status)

FOR all remaining columns set as their corresponding data type default
 varchar or char = ' ' (NOTE if column name ends in _des set to 'N/A'),
 date = '01/01/1900',
 float = 0
 number = 0

Insert Property (table name workout.subj_property)

internal_id_nbr = same as used in Insert Loan

addr1_txt = addr1_txt returned from REO

addr12_txt = addr2_txt returned from REO

city = city returned from REO

state = state returned from REO

zip_cd = zip cd returned from REO

occupancy_des = occupancy_des returned from REO

FOR all remaining columns set as their corresponding data type default
 varchar or char = ' ' (NOTE if column name ends in _des set to 'N/A'),
 date = '01/01/1900',
 float = 0
 number = 0

Application Data

'Note – this data will need to be accessed throughout the application

CertNbr	as string	'data entered on MainPage
lenderLoanNbr	as string	'data entered on MainPage
requestedWO	as string	'selected from listbox on MainPage
applSecUserID	as string	'passed in URL to fast track app
applSecOrgID	as string	'passed in URL to fast track app

Help Text - display in Status Window

- Certificate Number - 'Please enter a 10 digit GE Certificate Number'
- Lender Loan # - 'Please enter your Loan Number'
- Workout Type – 'Please select desired Workout'

REPAY / FORBEAR

Form Layout

see prototype repay.htm

Form Components

see screenFields.xls workbook, Repay worksheet for more details

REO/RMG Data

- Display Only Controls
 - Borrower Name, Certificate Number, Lender Loan Number, Requested Workout Name (Repay / Forbear)
 - Coverage Pct
- Prefilled and Modifiable text controls (prefilled from REO Extract data – see GetOrigAndDelqData in specMainPage.doc)
 - Primary Borrower First Name, MI, Last Name
 - CoBorrower First Name, MI, Last Name
 - Other Borrower First Name, MI, Last Name
 - Borrower Mailing Address Line 1
 - Line 2, City, ST and Zip
 - Property Address Line 1
 - Line 2, City ST and Zip
 - Investor Name
 - Bankruptcy Status
 - Current UPB
 - F/C Start Date, F/C Sale Date

SERVICER DATA ENTRY – color code as ????? Request is to distinguish these data entry fields from those prefilled by GE .

- Expected Repay Term (in Months)
- Expected Cure Date
- Has Loan Currently Been Referred to F/C

Form Links

NOTE - buttons should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

see spreadsheet screenFields.xls Repay worksheet

User Pushes Continue button

‘NOTE – all msgs defined below should be alert messages and should return the user to the same page, and for specific data elements return focus to those elements.

‘check data that GE returned from REO extract and that user may have modified

Call ValidateGEModifiableData

TOP SECRET - CONFIDENTIAL

'validate user entered data

'validate Expected Cure Date

If Expected Cure Date from screen is not a valid date then

msg 'Expected Cure Date is invalid'

End if

'check repay term

If repayTerm from screen > fastTrackMaxRepayTerm then

msg 'Term Entered Exceeds Maximum Allowable Term of 18 mos. Please Reduce The Term Or Push Continue To See Other Options.'

End if

'foreclosure has started so set radio button value to yes (if date is greater than Oracle default for

' "blank" date

If FCStartDate > '1900-01-01' then

referredToFCOpt = True

Else

If referredToFCOpt = True Then

If FCStartDate = '01/01/1900' then

msg 'Loan has been referred for foreclosure. Please enter a Foreclosure Start Date.'

End If

If FCSaleDate = '01/01/1900' then

msg 'Loan has been referred for foreclosure. Please enter a Foreclosure Sale Date.'

End If

End If

end if

see screenfields.xls Repay worksheet Input Validation for additional details

If no errors from data edits then

' call ValidWORequest function to run workout edits

If ValidWORequest Then

'check for insert or update by selecting from loans table (see logic in specsMainPage.doc for setting value of internal_id_nbr)

UpdateLoan see functions below

UpdateBorrower

UpdateSubjProperty

InsertRepay

Call SetApprovalConditions(requestedWO) (see applFunctions.doc)

Load ConditionalApproval screen

End If

End if

If NOT ValidWORequest then

load Does Not Meet Guidelines form with errReason (see ValidWORequest in specsApplFunctions for details on the various errReason values)

End If

Functions

Function ValidateGEModifiableData

FOR RELEASE

```

modifiedDataOK = TRUE
select state from state where @BorrAddrZipCode between begin_zip_cd and end_zip_cd

If (NO row returned) or ( state from DB <> state entered on screen) then
    msg 'Borrower Mailing Address has Invalid State and Zip combination'
    modifiedDataOK = FALSE
end if

'validate Property Mailing state and zip
'use workout.state table to check against
select state from state where @PropAddrZipCode between begin_zip_cd and end_zip_cd
If (NO row returned) or (state from DB <> state entered on screen) then
    msg 'Property Address has Invalid State and Zip combination'
    modifiedDataOK = FALSE
end if

'validate F/C start date
If FCStartDate not a valid date then
    msg 'F/C Start Date is invalid'
    modifiedDataOK = FALSE
end if

'validate F/C Sale Date
If FCSaleDate not a valid date then
    msg 'F/C Sale Date is invalid'
    modifiedDataOK = FALSE
End if

'validate Current UPB
If currUPB from screen > 0 then
    estCurrUPB = (origUPBAmt - (# mos difference between loanDueDate and OrigDate *
        P&I)
    minEstCurrUPB = estUPB - (estUPB * fastTrackUPBLowLim)
    maxEstCurrUPB = estUPB - (estUPB * fastTrackUPBMaxLim)
    If (minEstCurrUPB > CurrUPB from screen) Or (CurrUPB from screen >
        maxEstCurrUPB) then
        msg 'Current UPB is ????'
        modifiedDataOK = FALSE
    End If
End If

'check Investor Name
If investorName entered on MainPage screen <> InvestorName from REO extract then
    set investorLoanNbr to spaces
End if

'check Bankruptcy Status
If CHP7_BANKRUPTCY_IND = 'Y' then
    If BK_RLSE_DATE IS POPULATED then
        If BankruptcyStatus selected by the user is NOT = either 'Chp7 Reaffirmed
            Discharged' or 'Chp7 NonReaffirmed Discharged' then
            msg 'Our records show a release Chp 7 Bankruptcy. Please choose Chp7
                Reaffirmed Discharged or Chp 7 NonReaffirmed Discharged'
        End If
    Else

```


Selected value should be set to 'Chp 7 Active' as returned from REO
 Disable Bankruptcy Status (keep visible but do not allow it to be changed)
 End If

End If

End Function

UpdateLoan (table Name workout.loans)

Do DBGetFromKey using oLoans.internal_id_nbr

Update the following cols on the loans row

workout_cd = 'Repay Plan'

loan_due_date = Loan Due Date from Verified Data portion of the Form

loan_due_date_des = 'System'

curr_upb_amt = curr UPB Amt from Verified Data portion of the Form

curr_upb_amt_des = 'System'

override_fclsr_date = F/C Start Date data entered in Verified Info part of Form

reactive_workout_req_des = '???'

wo_approved_ind = 'P'

wo_status_ind = 'P'

investor_loan_nbr = returned from REO or data entered in Verified Portion of Form

fclsr_sale_date = F/C Sale Date from Verified Info portion of Form

Do DBStore to replace entire row

UpdateBorrower (table Name workout.borrowers)

**** REPEAT FOR EACH BORROWER RETURNED FROM THE REO EXTRACT FOR A
 MAX OF 3 BORROWERS

Do DBGetFromKey using oLoans.internal_id_nbr, borr_cd = 'PRI'

Update the following cols

last_name = from Primary Borrower Name on Verified Portion of Screen

first_name = from Primary Borrower Name on Verified Portion of Screen

middle_init = from Primary Borrower Name on Verified Portion of Screen

chapter7_ind = IF Bankruptcy Status selected by user on Verified Info part of Form = 'Chp 7

NonReaffirmed Discharged' or 'Chp 7 Reaffirmed Discharged' or 'Chp 7 Active' then set = 'Y' else set =
 'N'

chapter7_des = If chapter7_ind = 'Y' then set = 'System' else set = 'N/A'

borr_addr_txt1 = from Primary Borrower Addr on Verified Portion of Screen

borr_addr_txt2 = from Primary Borrower Addr on Verified Portion of Screen

borr_addr_city = from Primary Borrower Addr on Verified Portion of Screen

borr_addr_state = from Primary Borrower Addr on Verified Portion of Screen

borr_addr_zip = from Primary Borrower Addr on Verified Portion of Screen

servicer_bkrpt_status_ind = (set to 'R' if bankruptcy status selected = 'Chp7 Reaffirmed

Discharged. Set to 'N' if bankruptcy status selected = 'Chp7 NonReaffirmed Discharged' else set =
 'blank')

other_bankruptcy_des = (if Bankruptcy Status selected by user on Verified Info part of Form =
 'Chp 11Active' or 'Chp 13 Active' then set to 'Chapter 11' or 'Chapter 13' accordingly.)

Do DBStore to replace entire row

Do DBGetFromKey using oLoans.internal_id_nbr, borr_cd = 'COB'

Update the following cols

last_name = from Primary Borrower Name on Verified Portion of Screen

first_name = from Primary Borrower Name on Verified Portion of Screen

middle_init = from Primary Borrower Name on Verified Portion of Screen

Do DBStore to replace entire row

097894 "097894"

Do DBGetFromKey using oLoans.internal_id_nbr, borr_cd = 'OTH'

Update the following cols

last_name = from Primary Borrower Name on Verified Portion of Screen

first_name = from Primary Borrower Name on Verified Portion of Screen

middle_init = from Primary Borrower Name on Verified Portion of Screen

Do DBStore to replace entire row

Insert Property (table name workout.subj_property)

Call CheckREOToEnteredPropAddr

Do DBGetFromKey using oLoans.internal_id_nbr

Update the following cols

addr1_txt = property address line 1 from Verified Data on Form

addr12_txt = property address line 2 from Verified Data on Form

city = property address city from Verified Data on Form

state = property address state from Verified Data on Form

zip_cd = property address zip cd from Verified Data on Form

Insert Repay (table name workout.repay)

internal_id_nbr = same as used in Insert Loan

term_w_curr_disp_amt = RepayTerm (in Months) data entered on Form

workout_feasible_ind = 'Y'

estimated_cure_date = Expected Cure Date from screen (calculated)

FOR all remaining columns set as their corresponding data type default

varchar or char = ' ' (NOTE if column name ends in _des set to 'N/A'),

date = '01/01/1900',

float = 0

number = 0

CheckREOToEnteredPropAddr

'If the user changes any portion of the Subj Property address from what is shown as in REO

'then a warning message is inserted into Message_info table

'all Fast Track warning messages (err code 700series) will be included in the email to the WORep

If (property address line 1 from Verified Data on Form <> REO Extract addr1_txt) OR

(property address line 2 from Verified Data on Form <> REO Extract addr2_txt) OR

(property address city from Verified Data on Form <> REO Extract city) OR

(property address state from Verified Data on Form <> REO Extract state) OR

(property address zip cd from Verified Data on Form <> REO Extract zip_cd) Then

Insert into message_info values (@sysDateTime, @siteID, @certNbr, 'FASTTRACK',
'702', ' ', ' ')

End If

Help Text - display in Status Window

when the data entry field is clicked , display the following in the status window

Borrower First Name	show	Primary Borrower First Name
Borrower MI	show	Primary Borrower Middle Initial
Borrower Last Name	show	Primary Borrower Last Name
CoBorrower First Name	show	Primary Borrower First Name
CoBorrower MI	show	Primary Borrower Middle Initial
CoBorrower Last Name	show	Primary Borrower Last Name
Other Borrower First Name	show	Primary Borrower First Name

RECEIVED 09/10/01

Other Borrower MI	show	Primary Borrower Middle Initial
Other Borrower Last Name	show	Primary Borrower Last Name
Borrower Mailing Address Line 1	show	Current Mailing Address for Primary Borrower
Property Address	show	Valid Address for the insured property
Current UPB	show	Unpaid Principal Balance as of the last paid installment
F/C Start Date	show	Date when foreclosure started
F/C Sale Date	show	Expected foreclosure sale date
Investor Name	show	Choose appropriate investor
Bankruptcy Status	show	Choose appropriate bankruptcy status. Is not bankrupt, choose Not Applicable.
Expected Repay Term	show	Number of Months to be brought current under the repayment plan
Expected Cure Date	show	Date on which the loan will be brought current
Loan Referred to F/C	show	Check Yes if loan has been referred for foreclosure

09/26/2014 09:46:01

MOD

Form Layout

see prototype mod.htm

Form Components

REO/RMG Data

- Display Only Controls
 - Certificate Number, Lender Loan Number, Requested Workout Name (Repay / Forbear)
 - Coverage Pct
- Prefilled and Modifiable text controls
 - Primary Borrower First Name, MI, Last Name
 - CoBorrower First Name, MI, Last Name
 - Other Borrower First Name, MI, Last Name
 - Borrower Mailing Address Line 1
 - Line 2, City, ST and Zip
 - Property Address Line 1
 - Line 2, City ST and Zip
 - Investor Name
 - Bankruptcy Status
- Prefilled and Modifiable Credit Data controls
 - Current UPB
- Prefilled and Modifiable Date controls
 - F/C Start Date, F/C Sale Date

SERVICER DATA ENTRY – color code as ?????

Form Links

Note tabs should be for links to
 GE MI Home
 Contact Us
 Fast Track Workouts
 e-Wards
 Feedback

Form Processes

Load Form

see spreadsheet screenFields.xls MOD worksheet

User Pushes Continue button

‘NOTE – all msgs defined below should be alert messages and should return the user to the same page, and for specific data elements return focus to those elements.

‘check data that GE returned from REO extract and that user may have modified

Call ValidateGEModifiableData (see specsRepay.doc for function code)

‘validate user entered data

‘validate New Interest Rate

If New Interest Rate is < 1.000 or > 99.999 then
 msg 'New Interest Rate is invalid'
 End if

'New Term
 If New Term from screen > 360 then
 msg 'New Term cannot be greater than 360 months.'
 End if

'Effective Date
 ' "blank" date
 If Effective Date must be > '1900-01-01' and must be a valid date then
 msg 'Effective Date is invalid.'
 end if

see screenfields.xls MOD worksheet Input Validation for additional details

If no errors from data edits then
 ' call ValidWORequest function to run workout edits
 If ValidWORequest Then
 'check for insert or update by selecting from loans table (see logic in specsMainPage.doc for
 setting value of internal_id_nbr)
 UpdateLoan see functions below
 UpdateBorrower
 UpdateProperty
 InsertMOD
 Call SetApprovalConditions(requestedWO) (see applFunctions.doc)
 Load ConditionalApproval screen
 End If
 End if

If NOT ValidWORequest then
 load Does Not Meet Guidelines form with errReason (see ValidWORequest in
 specsApplFunctions for details on the various errReason values)
 End If

Functions

Function IsRateTypeMOD(typeMODWorkout As String)
 'this function will be used to indentify types of Modification workouts requiring a 'Interest Rate
 'use the entry selected in the Type of Modification dropdown on the MOD screen to 'call this function

 If (typeMODWorkout = 'MOD Rate' or typeOfModification = 'MOD RateTerm' or
 typeOfModification = 'MOD CapRateTerm' or typeOfModification = 'MOD RateCap' then
 return TRUE
 Else
 return FALSE
 End If
 End Function

Function IsTermTypeMOD(typeMODWorkout As String)
 'this function will be used to indentify types of Modification workouts requiring a New Term
 'use the entry selected in the Type of Modification dropdown on the MOD screen to 'call this function

 If (typeMODWorkout = 'MOD Term' or typeOfModification = 'MOD RateTerm' or

```

typeOfModification ='MOD CapRateTerm' or typeOfModification = 'MOD CapTerm' then
    return TRUE
Else
    return FALSE
End If
End Function

```

UpdateLoan

```

see specsRepayForbear.doc function section for code
PLUS
workout_cd = value selected in the Type of Modification dropdown on the MOD screen
capitalization_amt = Capitalization Amount from MOD Form
new_interest_rate = New Interest Rate from MOD Form
new_term = New Term from MOD Form
mod_effective_date = Effective Date from MOD Form

```

UpdateBorrower

```

see specsRepayForbear.doc function section for code

```

UpdateProperty

```

see specsRepayForbear.doc function section for code

```

InsertMOD (table name workout.mods)

```

internal_id_nbr = same as used in Insert Loan
mod_type_cd = set as CAP, CAPRATETERM, CAPTERM, RATE
RATECAP, RATETERM, TERM depending on the value selected in the Type Of Mod Dropdown on the
Mod Form
workout_feasible_ind = 'Y'

FOR all remaining columns set as their corresponding data type default
varchar or char = ' ' (NOTE if column name ends in _des set to 'N/A'),
date = '01/01/1900',
float = 0
number = 0

```

Help Text - display in Status Window

when the data entry field is clicked , display the following in the status window

for details see screenfields.cls MOD worksheet Help Text in Status Window column

097831 021604

BAP

Form Layout

see prototype bap.htm

Form Components

REO/RMG Data

- Display Only Controls
 - Certificate Number, Lender Loan Number, Requested Workout Name (Repay / Forbear)
 - Coverage Pct
- Prefilled and Modifiable text controls
 - Primary Borrower First Name, MI, Last Name
 - CoBorrower First Name, MI, Last Name
 - Other Borrower First Name, MI, Last Name
 - Borrower Mailing Address Line 1
 - Line 2, City, ST and Zip
 - Property Address Line 1
 - Line 2, City ST and Zip
 - Investor Name
 - Bankruptcy Status
- Prefilled and Modifiable Credit Data controls
 - Current UPB
- Prefilled and Modifiable Date controls
 - F/C Start Date, F/C Sale Date

SERVICER DATA ENTRY – color code as ?????

Form Links

Note tabs should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

see spreadsheet screenFields.xls BAP worksheet

User Pushes Continue button

‘NOTE – all msgs defined below should be alert messages and should return the user to the same page, and for specific data elements return focus to those elements.

‘check data that GE returned from REO extract and that user may have modified

Call ValidateGEModifiableData (see specsRepayForbear.doc for function code)

‘validate user entered data

‘validate Reinstatement Amount

T O T A L 0 9 0 4

If category_cd from REO is <> '00' and category_cd <> 'ND' And reinstatement_amt <=0 then
 msg 'Reinstatement Amount must be > \$0.'
 End if

see screenfields.xls BAP worksheet Input Validation for additional details

If no errors from data edits then
 ' call ValidWORequest function to run workout edits
 If ValidWORequest Then
 'check for insert or update by selecting from loans table (see logic in specsMainPage.doc for
 setting value of internal_id_nbr)
 UpdateLoan see functions below
 UpdateBorrower
 UpdateSubjProperty
 InsertBAP
 Call SetApprovalConditions(requestedWO) (see applFunctions.doc)
 Load ConditionalApproval screen
 End If
 End if

If NOT ValidWORequest then
 load Does Not Meet Guidelines form with errReason (see ValidWORequest in
 specsApplFunctions for details on the various errReason values)
 End If

Functions

UpdateLoan
 see specsRepayForbear.doc function section for code
 PLUS
 reinstatement_amt = Reinstatement Amount from BAP Form
 reinstatement_amt_des = 'System'
 bap_advance_amt = maxBAPAllowedAmt (as calculated in specsApplFunctions.doc Case BAP)

UpdateBorrower
 see specsRepayForbear.doc function section for code
 PLUS
 initial_cntrb_amt = Borrower Contribution from BAP Form where borr_cd = 'PRI' ONLY

UpdateProperty
 see specsRepayForbear.doc function section for code
 PLUS
 as_is_value_amt = Property Value amt from BAP Form
 as_is_value_amt_des = 'System'
 date_of_value_date = Property Value Date from BAP form
 occupancy_des = Selected value in Occupancy dropbox on the BAP Form

Insert BAP (table name workout.bap)

internal_id_nbr = same as used in Insert Loan
 disp_inc_used_amt = Positive Disposable Income from BAP Form
 max_bap_amt = maxBAPAllowedAmt as calculated in ValidWORequest Case BAP
 cntrb_to_max_bap_ind = If Borrower Contribution from BAP form > 0 then set to 'Y' else set to

'N'

09766136 001601

workout_feasible_ind = 'Y'

FOR all remaining columns set as their corresponding data type default
 varchar or char = ' ' (NOTE if column name ends in _des set to 'N/A'),
 date = '01/01/1900',
 float = 0
 number = 0

Help Text - display in Status Window

when the data entry field is clicked , display the following in the status window

see screenfields.xls BAP worksheet column Help Text in Status Window

BAPMOD

Form Layout

see prototype bapmod.htm

Form Components

REO/RMG Data

- Display Only Controls
 - Certificate Number, Lender Loan Number, Requested Workout Name (Repay / Forbear)
 - Coverage Pct
- Prefilled and Modifiable text controls
 - Primary Borrower First Name, MI, Last Name
 - CoBorrower First Name, MI, Last Name
 - Other Borrower First Name, MI, Last Name
 - Borrower Mailing Address Line 1
 - Line 2, City, ST and Zip
 - Property Address Line 1
 - Line 2, City ST and Zip
 - Investor Name
 - Bankruptcy Status
- Prefilled and Modifiable Credit Data controls
 - Current UPB
- Prefilled and Modifiable Date controls
 - F/C Start Date, F/C Sale Date

SERVICER DATA ENTRY – color code as ?????

Form Links

Note tabs should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

see spreadsheet screenFields.xls BAPMOD worksheet

User Pushes Continue button

‘NOTE – all msgs defined below should be alert messages and should return the user to the same page, and for specific data elements return focus to those elements.

‘check data that GE returned from REO extract and that user may have modified

Call ValidateGEModifiableData (see specsRepay.doc for function code)

‘validate user entered data

‘validate New Interest Rate

If New Interest Rate is < 1.000 or > 99.999 then
 msg 'New Interest Rate is invalid'
 End if

'New Term
 If New Term from screen > 360 then
 msg 'New Term cannot be greater than 360 months.'
 End if

'Effective Date
 ' "blank" date
 If Effective Date must be > '1900-01-01' and must be a valid date then
 msg 'Effective Date is invalid.'
 end if

'validate Reinstatement Amount
 If category_cd from REO is < '00' and category_cd < 'ND' And reinstatement_amt <=0 then
 msg 'Reinstatement Amount must be > \$0.'
 End if

see screenfields.xls BAPMOD worksheet Input Validation for additional details

If no errors from data edits then
 ' call ValidWORequest function to run workout edits
 If ValidWORequest Then
 'check for insert or update by selecting from loans table (see logic in specsMainPage.doc for
 setting value of internal_id_nbr)
 UpdateLoan see functions below
 UpdateBorrower
 UpdateSubjProperty
 InsertBAPMOD
 Call SetApprovalConditions(requestedWO) (see specsApplFunctions.doc)
 Load ConditionalApproval screen
 End If
 End if

If NOT ValidWORequest then
 load Does Not Meet Guidelines form with errReason (see ValidWORequest in
 specsApplFunctions for details on the various errReason values)
 End If

Functions

UpdateLoan
 see specsRepayForbear.doc function section for code
 PLUS
 workout_cd = 'BAP' & value selected in Type of Modification dropbox on the BAPMOD screen.
 (ex. BAPMOD CapRateTerm)
 reinstatement_amt = Reinstatement Amount from BAPMOD Form
 reinstatement_amt_des = 'System'
 bap_advance_amt = maxBAPAdvanceAmt (as calculated in specsApplFunctions.doc Case BAP)
 capitalization_amt = Capitalization Amount from MOD Form
 new_interest_rate = New Interest Rate from MOD Form
 new_term = New Term from MOD Form
 mod_effective_date = Effective Date from MOD Form

BAPMOD CapRateTerm

UpdateBorrower

see specsRepayForbear.doc function section for code

PLUS

initial_cntrb_amt = Borrower Contribution from BAPMOD Form where borr_cd = 'PRI' ONLY

UpdateProperty

see specsRepayForbear.doc function section for code

PLUS

as_is_value_amt = Property Value amt from BAPMOD Form

as_is_value_amt_des = 'System'

date_of_value_date = Property Value Date from BAPMOD form

occupancy_des = Selected value in Occupancy dropbox on the BAPMOD Form

Insert BAPMOD (table name workout.bap_payment)

internal_id_nbr = same as used in Insert Loan

disp_inc_used_amt = Positive Disposable Income from BAPMOD Form

bapmod_type_cd = set as CAP, CAPRATETERM, CAPTERM, RATE

RATECAP, RATETERM, TERM depending on the value selected in the Type Of Mod Dropbox on the BapMod Form

max_bap_amt = maxBAPAdvanceAmt as calculated in ValidWOREquest Case BAP

cntrb_to_max_bap_ind = If Borrower Contribution from BAP form > 0 then set to 'Y' else set to

'N'

cap_ge_insure_amt = maxCAPGEInsure as calculated in ValidWOREquest Case BapMod (see

specsApplFunctions.doc)

workout_feasible_ind = 'Y'

FOR all remaining columns set as their corresponding data type default

varchar or char = ' ' (NOTE if column name ends in _des set to 'N/A'),

date = '01/01/1900',

float = 0

number = 0

Help Text - display in Status Window

when the data entry field is clicked , display the following in the status window

see screenfields.xls BAPMOD worksheet column Help Text in Status Window

03/28/2010 10:46:04

DEED-IN-LIEU

Form Layout

see prototype DIL.htm

Form Components

REO/RMG Data

- Display Only Controls
 - Certificate Number, Lender Loan Number, Requested Workout Name (Repay / Forbear)
 - Coverage Pct
- Prefilled and Modifiable text controls
 - Primary Borrower First Name, MI, Last Name
 - CoBorrower First Name, MI, Last Name
 - Other Borrower First Name, MI, Last Name
 - Borrower Mailing Address Line 1
 - Line 2, City, ST and Zip
 - Property Address Line 1
 - Line 2, City ST and Zip
 - Investor Name
 - Bankruptcy Status
- Prefilled and Modifiable Credit Data controls
 - Current UPB
- Prefilled and Modifiable Date controls
 - F/C Start Date, F/C Sale Date

SERVICER DATA ENTRY – color code as ?????

Form Links

Note tabs should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

see spreadsheet screenFields.xls DIL worksheet

User Pushes Continue button

‘NOTE – all msgs defined below should be alert messages and should return the user to the same page, and for specific data elements return focus to those elements.

‘check data that GE returned from REO extract and that user may have modified

Call ValidateGEModifiableData (see specsRepayForbear.doc for function code)

‘validate user entered data

‘count the number of borrowers selected in the Identify Deceased Borrower dropbox

nbrOfDeceasedBorrowers = count of all selected entries in dropbox

09/03/04 09:04:04

see screenfields.xls DIL worksheet Input Validation for additional details

If no errors from data edits then

‘ call ValidWORequest function to run workout edits

If ValidWORequest Then

‘check for insert or update by selecting from loans table (see logic in specsMainPage.doc for setting value of internal_id_nbr)

UpdateLoan (DBStore) see functions below

UpdateBorrower (DBStore)

UpdateProperty (DBStore)

InsertDIL (DBStore)

Call SetApprovalConditions(requestedWO) (see applFunctions.doc)

Load ConditionalApproval screen

End If

End if

If NOT ValidWORequest then

load Does Not Meet Guidelines form with errReason (see ValidWORequest in specsApplFunctions for details on the various errReason values)

End If

Functions

UpdateLoan

see specsRepayForbear.doc function section for code

PLUS

workout_cd = ‘Deed-in-lieu’

est_payoff_date = Payoff Date from DIL Form

actual_payoff_amt = Payoff Amount from DIL Form

actual_payoff_amt_des = ‘System’

UpdateBorrower

see specsRepayForbear.doc function section for code

PLUS

deceased_ind = ‘Y’ for each Borrower selected by the user in the Identify Deceased Borrowers dropbox.

UpdateProperty

see specsRepayForbear.doc function section for code

PLUS

as_is_value_amt = Property Value amt from DIL Form

as_is_value_amt_des = ‘System’

date_of_value_date = Property Value Date from DIL form

Insert DIL (table name workout.dil)

internal_id_nbr = same as used in Insert Loan

ge_loss_overguar_amt = lossOverGuaranteeAmt as calculated in ValidWORequest Case DIL (see specsApplFunctions.doc for more information)

workout_feasible_ind = ‘Y’

FOR all remaining columns set as their corresponding data type default

varchar or char = ‘ ‘ (NOTE if column name ends in _des set to ‘N/A’),

date = ‘01/01/1900’,

```
float = 0
number = 0
```

Help Text - display in Status Window

when the data entry field is clicked , display the following in the status window

see screenfields.xls DIL worksheet column Help Text in Status Window

Form Layout

see prototype [presale.htm](#)

Form Components

REO/RMG Data

- Display Only Controls
 - Certificate Number, Lender Loan Number, Requested Workout Name (Repay / Forbear)
 - Coverage Pct
- Prefilled and Modifiable text controls
 - Primary Borrower First Name, MI, Last Name
 - CoBorrower First Name, MI, Last Name
 - Other Borrower First Name, MI, Last Name
 - Borrower Mailing Address Line 1
 - Line 2, City, ST and Zip
 - Property Address Line 1
 - Line 2, City ST and Zip
 - Investor Name
 - Bankruptcy Status
- Prefilled and Modifiable Credit Data controls
 - Current UPB
- Prefilled and Modifiable Date controls
 - F/C Start Date, F/C Sale Date

SERVICER DATA ENTRY – color code as ?????

Form Links

Note tabs should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

see spreadsheet [screenFields.xls](#) Presale worksheet

User Pushes Continue button

‘NOTE – all msgs defined below should be alert messages and should return the user to the same page, and for specific data elements return focus to those elements.

‘check data that GE returned from REO extract and that user may have modified

Call `ValidateGEModifiableData` (see `specsRepayForbear.doc` for function code)

```
'validate user entered data'
```

If Payoff Date \diamond Presale ClosingDate then

msg 'Payoff Date must equal Closing Date.'
End If

see screenfields.xls PRESALE worksheet Input Validation for additional details

If no errors from data edits then
' call ValidWORequest function to run workout edits
 If ValidWORequest Then
 'check for insert or update by selecting from loans table (see logic in specsMainPage.doc for
setting value of internal_id_nbr)
 UpdateLoan see functions below
 UpdateBorrower
 UpdateProperty
 InsertPresale
 Call SetApprovalConditions(requestedWO) (see applFunctions.doc)
 Load ConditionalApproval screen
 End If
End if

If NOT ValidWORequest then
 load Does Not Meet Guidelines form with errReason (see ValidWORequest in
specsApplFunctions for details on the various errReason values)
End If

Functions

UpdateLoan
 see specsRepayForbear.doc function section for code
 PLUS
 workout_cd = 'Presale'
 est_payoff_date = Payoff Date from Presale Form
 actual_payoff_amt = Payoff Amount from Presale Form
 actual_payoff_amt_des = 'System'

UpdateBorrower
 see specsRepayForbear.doc function section for code
 PLUS
 initial_cntrb_amt = Borrower Cash Contribution from PRESALE Form where borr_cd = 'PRI'
ONLY

UpdateProperty
 see specsRepayForbear.doc function section for code
 PLUS
 as_is_value_amt = Property Value amt from PRESALE Form
 as_is_value_amt_des = 'System'
 date_of_value_date = Property Value Date from PRESALE form
 act_net_on_sale_amt = Net On Sale Amount as entered on Presale Form (only enabled for Chp 7
NonReaffirmed Bankruptcy Status)
 act_net_on_sale_amt_des = 'System'

Insert PRESALE (table name workout.presale)

 internal_id_nbr = same as used in Insert Loan
 workout_feasible_ind = 'Y'
 presale_closing_date = Presale closing Date as entered on Presale Form

Downloaded by [illegible]

Help Text - display in Status Window
when the data entry field is clicked , display the following in the status window

see screenfields.xls PRESALE worksheet column Help Text in Status Window

Does Not Meet Guidelines

Form Layout

see prototype fullde.htm

Form Components

- see screenfields.xls Generic Approval worksheet

Form Links

Note tabs should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

see spreadsheet screenFields.xls Fail Guidelines worksheet

Data passed to this screen should be:

proceedMsg text

error Code #

proceed Options (could be up to 3)

To get the errReason,

Select err_message_txt from err_codes where err_code = error Code # passed to screen

Insert into message_info values (current system date/time, userSiteID, certNbr, 'FASTTRACK', error Code #, ' ', ' ')

'Update the loans table. Select the row from the loans table for this cert using DBGetFromKey and then

Update the wo_approved_ind = 'D', wo_complete_date = current system date

Put the row back in the DB with DBSTORE

Build the screen to show:

Error Reason

Proceed Message

Proceed Options

User does 'Click Here' for another workout

*** the following is on hold pending Marketing's Project on bringing eWards internal

Get webbucks info for certaLinkUserID.

Set webbucks = current web bucks amount + FastTrackFailedWOeWard

User does 'Click Here' for Full Data Entry

*** the following is on hold pending Marketing's Project on bringing eWards internal

Get webbucks info for certaLinkUserID.

Set webbucks = current web bucks amount + FastTrackFullDEWOeWard

09/29/01 04:01

User does 'Click Here' for Submit Package

*** the following is on hold pending Marketing's Project on bringing eWards internal

Get webbucks info for certaLinkUserID.

$$\text{Set webbucks} = \text{current web bucks amount} + \text{FastTrackSubmitWOeWard}$$

Approval

Form Layout

see prototype BAPapproval.htm

NOTE - The Promissory Note controls will only apply to BAP, BAPMOD and Presales

Form Components

- see screenfields.xls Generic Approval worksheet

Form Links

Note tabs should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

see spreadsheet screenFields.xls Generic Approval worksheet

NOTE – For requestedWO = BAP, BAPMOD or Presale with Promissory Note Amount
> 0 then

place labels, single select dropbox, start date and button on html form

‘call the following to get firstPaymentDate to display on screen

Call CalcNoteStartDate

show approval conditions as determined by the

Call SetApprovalConditions(requestedWO) for each type of workout (see individual workout
specs and specsApplFunctions.doc for more info)

Show all other controls as disabled EXCEPT the ‘Accept This Workout’ checkbox which should
always load as unchecked.

User Checks ‘Accept This Workout’ checkbox ON

enable Print Approval Letter control and Promissory Note footer (IF APPLICABLE)

User Pushes ‘Continue’ button

If ‘Accept This Workout’ checkbox is not checked on then

Alert Msg ‘Please check Accept This Workout checkbox ON before proceeding or push Cancel
This Workout button.’

End If

Select the loans row from the database for this internal_id_nbr

Update the wo_approved_ind = ‘A’, wo_complete_date = current system date

Put the row back in the DB with DBSTORE

Call UpdateREOWOStatus

*** the following is on hold pending Marketing’s Project on bringing eWards internal

Get webbucks info for certaLinkUserID.

Set webbucks = current web bucks amount + FastTrackApprovedWOeWard

User Pushes the Print Approval Letter button

formName will be determined by workout request type. Once forms are created and named then this portion of the spec can be completed.

User Pushes Print Note button

‘this button and associated controls will only exist on the form for an approval on a BAP or BAPMOD or a Presale with contribution in the form of promissory note ONLY.

Call `calcPromNoteData`

If workoutType = 'Presale' then

Call UpdatePresaleNoteInfo

Else

If workout type = 'BAP' then

Call UpdateBAPNoteInfo

Else

Call UpdateBAPMODNoteInfo

End If

End If

Call NOVA passing (internal_id_nbr, certaLinkUserID, eLMONovaID, eLMONovaPswd, DocumentName, stored procedure name?????)

NOTE – all data for the document will be retrieved via a stored procedure

Check with Allison on how to handle the call to populate the PDF document.

NOTE – attached to Promissory Note document will be Borrower Assistance Plan Authorization and GE Promissory Note Checklist as separate pages in the single document. This way, users will request to print once but get all 3 documents printed.

User Pushes Continue/Go Button

Load ThankYou form (see screenfields.xls Thank You worksheet for details on form load)

Functions

CalcNoteStartDate

If requestedWO = 'Presale' then

tempDate = compute as the first day of the next month from

oPresale.presale closing date month.

(e.g. oPresale.presale_closing_date = 10/24/00 then tempDate = 11/01/2000
oPresale.presale_closing_date = 12/12/00 then tempDate = 01/01/2001)

```
firstPaymentDate = tempDate + 2 months
```

End If

```

If requestedWO = 'BAP' OR
  requestedWO = 'BAPMOD Cap' OR
  requestedWO = 'BAPMOD CapRateTerm' or
  requestedWO = 'BAPMOD CapTerm' or
  requestedWO = 'BAPMOD Rate' or
  requestedWO = 'BAPMOD RateCap' or
  requestedWO = 'BAPMOD RateTerm' or
  requestedWO = 'BAPMOD Term' then
  tempDate = compute as the first day of the next month from oLoans.wo_complete_date
  firstPaymentDate = tempDate + 2 months
End if

```

CalcPromNoteData

```

If requestedWO = 'BAP' then
  NoteAmount = oBap.max_adv_amt
Else
  If requestedWO in ( 'BAPMOD Rate','BAPMOD Term','BAPMOD
    RateTerm','BAPMOD Cap','BAPMOD RateCap','BAPMOD CapTerm','BAPMOD
    CapRateTerm') then
    NoteAmount = obap_payment.max_adv_amt where bapmod_type_cd =
      requestedWO
  Else
    If requestedWO = 'Presale' then
      NoteAmount = oPresale.act_cntrb_pn_amt
    End If
  End If
End If

finalPaymentDate = firstPaymentDate + payment term # from the selected payment/term option in
the Payment / Term dropdown

LastPayment = payment term # from the selected payment/term option in the Payment / Term
dropdown

NumberPayments = LastPayment - 1

MonthlyPaymentAmt = payment amount from the selected payment / term option in the Payment
/ Term dropdown

tempPrincipalAmt = MonthlyPaymentAmt * LastPayment

FinalPaymentAmt = MonthlyPaymentAmt = (NoteAmount - tempPrincipalAmt)

UpdatePresaleNoteInfo
Do DBGetFromKey on table Presale using internal_id_nbr
Update the following fields with those selected from the dropdown:
  note_start_date          /*firstPaymentDate as calculated above
  note_term_nbr            /* term = nbr of months
  note_mo_payment_amt      /*monthly payment amount

Do DBStore to replace entire row

UpdateBAPNoteInfo
Do DBGetFromKey on table BAP using internal_id_nbr

```

Update the following fields with those selected from the dropbox:

note_start_date	/*firstPaymentDate as calculated above
note_term_nbr	/* term = nbr of months
note_mo_payment_amt	/*monthly payment amount

Do DBStore to replace entire row

UpdateBAPMODNoteInfo

Do DBGetFromKey on table BAP_PAYMENT using internal_id_nbr and bapmod_type_cd = BAP&type of mod workout selected (ex. BAPMOD Term, BAPMOD CapRateTerm, etc.)

Update the following fields with those selected from the dropbox:

note_start_date	/*firstPaymentDate as calculated above
note_term_nbr	/* term = nbr of months
note_mo_payment_amt	/*monthly payment amount

Do DBStore to replace entire row

UpdateREOWOStatus

- will need to pass cert_nbr and master_policy_nbr to identify the record in the REO IDMS Database
- will generate a CICS transaction , connect to CICS through the black box and update specific workout status fields in the REO system, realtime. See specsREOUpdate.doc for more information)

REO IDMS Database

Submit Package (aka Contact Us)

Form Layout

see prototype ContactUs.htm

Form Components

- see screenfields.xls Submit Package worksheet

Form Links

Note tabs should be for links to

GE MI Home

Contact Us

Fast Track Workouts

e-Wards

Feedback

Form Processes

Load Form

GetEmailAccount for WO Rep

Select user_id from servicer_info where servicer_org_id = certaLinkOrgID (see specsMainPage2.doc for more info on certaLinkOrgID)

Get associated eMail account for this user

(note user_id = domain id ?? how to get to email address)

??? Get NLWC central fax number for Fast Track users

see spreadsheet screenFields.xls Submit Package worksheet

User does 'Click Here To Email Your GE Workout Rep'

Check with Kristen on how this works

REO Extract

Description:

This program will extract specific REO data by certificate number from both the Certfile and REO. There are several data edits which are required to prevent a user from doing a Fast Track workout on a certificate with specific conditions – these can only be worked here in GE. If any of the edits fail, the program will generate an error record. Both file formats, the data record and the error record, can be found on

p:\misso\workout\le-lmo dmadv\design\REOextractIntrfc.xls

The calling web application will generate this request for one certificate at a time.

Input Transaction:

transactionName	???standards???
cert_nbr	char(10) as data entered by user
delq_org_id	char(20) passed to Fast Track from login at servicer page in URL
investor_code	Char(5) valid values are either 'FNMA ' or 'FHLMC'
lender_loan_nbr	char(20) as data entered by user

Databases:

copybooks needed for the following:

VSAM CertRec

Get Various data elements (see REOextractIntrfc.xls)

IDMS g8-cert-rec

g8-delq-rec
g8-recovery-rec
get various data elements from these (see REOextractIntrfc.xls)

Duplicate cert-nbr table (used to get a master policy number when IDMS has more than one record with the same certificate number – only about 400 of these exist)

DB2 for investor info

```
SELECT INVESTOR_CODE
FROM REODBT01.REO_INVEST_WORKOUT
WHERE reodbt01.reo invest workout = @certNbr
```

TRANSLATE AS: IF INVESTOR_CODE = 'FNMA' THEN SET TO 'Fannie Mae' ELSE IF
INVESTOR_CODE = 'FHLMC' THEN SET TO 'Freddie Mac' ELSE SET TO 'N/A'

Output:

returned to calling web application via blackbox as a data stream (single string)

REO Extract Data Record

Error Record

Processing:

Initialize all fields in the data record and error record to their default values according to their data types

Edits:

FATAL ERRORS

If any of the following fatal errors occur while processing a cert number, do NOT write a record to the extract output file. Instead, return an error record defined as :

ERROR_DATETIME	char(18) format MM/DD/CCYY00:00:00
CERT_NBR	char(20)
ERROR_CODE	char(10)
ERROR_FIELD	char(40)
ERROR_DATA	char(60)

Please perform the edits in the following specified order

1. Users org id does not match the delq servicing org id in REO
 If G8-CERT-REC DELQ-SERV-ORG-KEY not = orgID of the user logged in (passed in with calling transaction)
 err code = 703
 err data = OrgID of user logged in
 err field = Prior Servicer Org Id
 End If
2. get Active / Passive Ind from reodbt01.reo_invest_workout, reodbt01.reo_lendr_name
 SELECT INVESTOR_CODE, WORKOUT_PGM_IND
 FROM REODBT01.REO_INVEST_WORKOUT,
 REODBT01.REO_LENDR_NAME
 WHERE (cert_nbr from input file) = CERT_NBR
 AND REO_INVEST_WORKOUT.LENDR_NAME_KEY =
 REO_LENDR_NAME.LENDR_NAME_KEY

 If (investor_code = ('FNMA ' and WORKOUT_PGM_IND = 'A')) or (investor_code =
 'FHLMC') then
 Error with Error-Code = 103
 Error-Field = '
 Error-Data = INVESTOR_CODE FROM
 REODBT01.REO_INVEST_WORKOUT

 LMO err_codes table message_txt for err_code_txt = 103 will be
 Message 'Loan is part of Active Delegated Program. Refer Workout to Investor.'

 3. If INACTIVE, error
 If VSAM CERTREC mg-reason-inact > ""
 error with Error-Code = 121
 Error-Field = '
 Error-Data = mg-reason-inact

 LMO err_codes table message_txt for err_code_txt = 121 will be
 Message 'Our records indicate the Insurance Certificate is Inactive. Please contact Policy Servicing
 – see Contact Us.'

 4. If Pool Insured, error
 If (3rd digit of cert_nbr = '5') Or (VSAM CERTREC MG-NBR-XREF-CERT has a 3rd digit =
 '5')
 error with Error-Code = 114
 Error-Field = '
 Error-Data = '

LMO err_codes table message_txt for err_code_txt = 115 will be
 Message 'This loan may not be processed through Fast Track. Please submit your completed package to GE.'

5. If the file location starts with a C# this indicates that the loan is in Claims – workout is not allowed
 If G8-CERT-REC first 2 positions of FILE-LOC = 'C#'
 (then certificate is in Claims)

Error with Error-Code = 118
 Error-Field = FILE-LOC
 Error-Data = ''

LMO err_codes table message_txt for err_code_txt = 118 will be
 Message Text 'Our records indicate that the lender may have acquired title. Please contact GE Claims Dept. for assistance – see Contact Us.'

6. If cated is 40 and lender acquired title date has a valid date in it then do not allow workout
 If g8-delq-rec mg-delq-category = 40 and g8-delq-rec len-acq-title-date-6 is populated then
 Error with Error-Code = 120
 Error-Field = ''
 Error-Data = value of len-acq-title-date-6

LMO err_codes table message_txt for err_code_txt = 120 will be
 Message Text 'Our records indicate that the lender may have acquired title. Please contact GE Claims Dept. for assistance – see Contact Us.'

7. If Special Deal for this certificate is a Claim Impact Special Deal Number (i.e. customer contract deal_category = 'C') and Indemnity Status = 'A' (again on customer contract) then request to submit package

NOTE – IF ANY SPECIAL DEAL NBR RETURNS WITH STATUS_FLAG = 'A' and
 SPDL_CATEGORY = 'C' THEN END THE EDIT CHECK WITH THE ERROR CODE below

NOTE – the CICS pgm G8CC013IX will be called with EACH special deal number. This program can also be called with a certificate number and so the data returned occurs 7 times.
 For this program, ONLY LOOK AT THE 1st OCCURRENCE.

If VSAM CERTREC MG-CODE-SPECIAL-DEAL not = Spaces or zero then

Call blackbox 'G8CC013IX' passing MG-CODE-SPECIAL-DEAL and setting SPDL-IND = 'S'

- *+ Check the status flag and the deal category to determine claim impact
 If X1-SPDL-STATUS-FLAG(1) = 'A' and X1-SPDL-CATEGORY(1) = 'C'

Error with Error-Code = 116
 Error-Field = 'IndemStat A'
 Error-Data = 'DealCat C'

End If

If no error from above, then go to the DB2 table to get any other special deal numbers associated with this certificate.

The SQL select statement to get the special deal #'s from DB2 is

EXEC SQL
 SELECT CLAIM_CODE_1, CLAIM_CODE_2, CLAIM_CODE_3, CLAIM_CODE_4,
 CLAIM_CODE_5

FOR RELEASE UNDER E.O. 14176

```
:LM-CLAIM-CODE-1
:LM-CLAIM-CODE-2
:LM-CLAIM-CODE-3
:LM-CLAIM-CODE-4
:LM-CLAIM-CODE-5
```

WHERE CERT_NBR = CERT_NBR PASSED IN TRANSACTION + G8-
CERT-REC MPOL-NBR
(NOTE - NEEDS TO BE A CHAR(20) DATA FIELD, 10 FROM CERT NBR AND 10
FROM MASTER POLICY NBR)

If LM-CLAIM-CODE-x (where x = 1 through 5) not = Spaces or zero then

*+ Check the status flag and the deal category to determine claim impact.

Error-Data = 'DealCat C'

Message Text ' This loan may not be processed through Fast Track. Please submit your completed package to GE.'

Error-Data = '470'

Message 'This loan may not be processed through Fast Track. Please submit your completed package to GE.'

Error-Data = '470'

Message 'This loan may not be processed through Fast Track. Please submit your completed package to GE.'

Error-Data = name of file/record with the physical DB error (see above list)

11. If a physical database error occurred attempting to read ONLY the following input database records
 ‘ACBCERX-Read’, ‘ACBCERX-Start’
 then
 error with Error-Code = 108
 Error-Field = ‘Database’
 Error-Data = name of file/record with the physical DB error
 i.e. ‘ACBCERX-Read’, ‘ACBCERX-Start’

LMO err_codes table message_txt for err_code_txt = 108 will be Message 'Physical Database Error Occurred'

REO Update

Functional Description

REO mainframe data will be updated from e-lmo Fast Track in a variety of situations. When a workout is approved in Fast Track, the REO system will be updated with the current workout information just as is manually done today when workouts are approved in LMO. In the case where a NonCure workout (i.e. Presale or Deed-in-lieu) has been approved on a non-delinquent certificate, a delinquent entry will be made in REO and then the appropriate workout approval information will be recorded. When a previously approved workout (as noted by the status code in REO) is failed in e-lmo Fast Track, REO will be updated to reflect this as well.

NOTE – ALL DATES WILL BE IN THE FORMAT MM/DD/CCYY -> INSURE THAT THE CENTURY IN REO DATES ARE SET AS PER Y2K GUIDELINES

Input Transaction Format:

Format for Approval

APPROVAL, cert_nbr, servicer_org_id, rep_initials, workout_type, fu_date, bankruptcy_status, fc_start_date, fc_sale_date, curr_upb_amt, loan_due_date, mail_addr1, mail_addr2, mail_city, mail_state, mail_zipcd, investor_name, workout_pgm_ind, investor_loan_nbr, servicer_name, lender_loan_nbr

Format for Investor Update

'INVCHANGE', cert_nbr, master_policy_nbr, REOInvestorName, workoutPgmInd, investorLoanNbr, 'F-EFT', systemDate)

Format for Failed

FAILED, cert_nbr, servicer_org_id

(??? Only changes eval and status, STATUS CHANGE DATE and lastUpdateDate -> any followup date setting/code??)

Format for NotepadUpdate

NOTEPAD, cert_nbr, servicer_org_id, rep_initials, workout_type, notepad_txt

Output Record Format

ERROR_IND	3
ERROR_DATETIME	18
CERT_NBR	10
ERROR_CODE	10
ERROR_FIELD	40
ERROR_DATA	65

P-MAIN-TRANSACTION PROCESSING.

```

/* DEPENDING ON THE REQUEST, THE LOGIC REQUIRES DATA TO BE UPDATED ON
/* G8-DELQ-REC, G8-SERVADDR-REC AND DB2 REO_INVEST_WORKOUT,
/* REO_LENDR_NAME TABLE
/* IF worout_type = 'PRESALE' or 'DEED-IN-LIEU' AND RECORD NOT FOUND IN REO FOR /*
THE CERTNBR PASSED IN ON THE TRANSACTION
/* THEN ONLY SETUP A DELQ
/* A NONDELINQUENT CERT

```

NOTE – FOR ANY OTHER DB ERRORS WHICH OCCUR OTHER THAN THOSE SPECIFIED BELOW,

Error with Error-Code = 108
 Error-Field = 'G8-DELQ-REC'
 Error-Data = ' '

LMO err_codes table message_txt for err_code_txt = 108 will be
 Message Text 'Physical Database Error Occurred – Please Contact GE for assistance.'
 Exit processing.

/* check for duplicate cert

Read ACBCERX for the cert_nbr from the transaction.

IF record found then

Read G8-CERT-REC where CM-MPOL-CERT-KEY = the cert_nbr + master policy nbr key from ACBCERX

IF G8-CERT-REC NOT FOUND

GET the NEXT ACBCERX record and use that cert_nbr + master policy nbr for the key
 Read G8-CERT-REC where CM-MPOL-CERT-KEY = the cert_nbr + master policy nbr key from ACBCERX

IF G8-CERT-REC NOT FOUND

IF WORKOUT_TYPE = 'PRESALE' OR

WORKOUT_TYPE = 'DEED-IN-LIEU'

PERFORM P-DELQ-REC-SETUP /*SEE BELOW FOR SPECS

END IF

ELSE

Create error record with

Error-Code = 107

Error-Field = ' '

Error-Data = ' '

Message txt for error code 107 'Record not found for cert'.

End Processing.

End If

Else

IF CM-DELQ-SERV-ORG-KEY = servicer_org_id from transaction then
 continue processing using this MPOL-CERT-KEY

ELSE

GET the next ACBCERX record

Read G8-CERT-REC where CM-MPOL-CERT-KEY = the cert_nbr + master policy nbr key from ACBCERX

IF G8-CERT-REC NOT FOUND

IF WORKOUT_TYPE = 'PRESALE' OR

WORKOUT_TYPE = 'DEED-IN-LIEU'

PERFORM P-DELQ-REC-SETUP

END IF

ELSE

Create error record with

Error-Code = 107

Error-Field = ' '

Error-Data = ' '

Message txt for error code 107 'Record not found for cert'.

End Processing


```

/* no duplicates for this cert nbr
  Read G8-CERT-REC using the cert_nbr passed in on the transaction
  If G8-CERT-REC NOT FOUND
    Create error record with
    Error-Code = 107
    Error-Field = ' '
    Error-Data = ' '
    Message txt for error code 107 'Record not found for cert'.
    End Processing.
  End If
END IF

/* NOW HAVE G8-CERT-REC so READ G8-DELQ-REC
  If G8-CERT-REC first 2 positions of FILE-LOC = 'C#' or 'C-'
  (then certificate is in Claims)
    Error with Error-Code = 118
    Error-Field = FILE-LOC
    Error-Data = ' '

    LMO err_codes table message_txt for err_code_txt = 118 will be
    Message Text 'This loan is not available for a Workout Approval. Please contact GE for
    assistance.'

    ??? what if file loc like 'I-xxx' for in investigation ???

  IF G8-DELQ-REC not found for cert then
    Error with Error-Code = 108
    Error-Field = 'G8-DELQ-REC'
    Error-Data = ' '

    LMO err_codes table message_txt for err_code_txt = 108 will be
    Message Text 'Physical Database Error Occurred - Please Contact GE for assistance.'
    Exit processing.
  End IF

  If transaction request is APPROVAL
    Perform P-IDMS-APPROVAL-UPDATE.
  If transaction request is INVCHANGE
    Perform P-DB2-UPDATE.
  If transaction request is FAILED
    Perform P-IDMS-FAILED-UPDATE.
  If transaction request is NOTEPAD
    Perform P-IDMS-NOTEPAD-INSERT.

  If no errors OTHER THAN '703' (which is a non-fatal error) then
    Create error record with
    Error-Code = 901
    Error-Field = ' '
    Error-Data = ' '
    Message txt for error code 901 'REO Successfully Updated'

  P-IDMS-APPROVAL-UPDATE.
  If G8-DELQ-REC found then continue processing as follows.

  If DL-MG-DELO-CATEGORY = '10' OR '20' OR '30'

```

```

IF LOAN_DUE_DATE >= CURRENT_DATE
/*      THEN LOAN IS CURRENT.  CHANGE CAT CD & CURE DATE
/*      DO NOT UPDATE LOAN DUE DATE OR CURR UPB
      DL-MG-DELQ-CATEGORY = '00'
      DL-PREV-WORKOUT-CODE =
          DL-CURR-WORKOUT-CODE
      DL-CURR-WORKOUT-CODE = ' '
      DL-CURED-DATE-6 = CURRENT DATE
/* INSURE CCYY IS SET AS PER Y2K GUIDELINES
DO NOTEPAD-ENTRY passing the following values
      CURRENT-DATE, 'F-EFT', 'CURE UPDATED THROUGH EFT'
ELSE
/* UPDATE CURRENT UNPAID BALANCE
DL-CURR-LOAN-BAL = CURR_UPB_AMT
/* UPDATE LOAD DUE DATE - HANDLE CENTURY SETTING
DL-LOAN-DUE-DATE-6 = LOAN_DUE_DATE
/* INSURE CCYY IS SET AS PER Y2K GUIDELINES
END IF
END IF

IF DL-MG-DELQ-CATEGORY = '00' OR 'ND' OR '10' OR '20' OR '30' THEN
  IF DL-WORKOUT-STAT-CODE = 'P'
    Error with Error-Code = 801
    Error-Field = 'WORKOUT TYPE'
    Error-Data = value of DL-CURR-WORKOUT-IND

    LMO err_codes table message_txt for err_code_txt = 801 will be
    Message 'Approved workout already exists on this certificate. Please contact GE for
    assistance - see Contact Us.'
  End If

/* BANKRUPTCY UPDATE - CHP, FILE DATE, RLSE DATE AND RELEASE TYPE
  IF BANKRUPTCY_STATUS = 'CHP 7 NONREAFFIRMED DISCHARGED' OR
    'CHP 7 REAFFIRMED DISCHARGED'
    DL-BKRPTCY-CH = '07'
    DL-BKRPTCY-FILE-DATE-6 = ?????
    DL-BKRPTCY-RLSE-DATE-6 = ????? /* INSURE CCYY IS HANDLED
    DL-BKRPTCY RELEASE TYPE = 'DC'

/* SET PREVIOUS WORKOUT TO CURRENT WORKOUT
  IF DL-CURR-WORKOUT_CODE > ' '
    DL-PREV_WORKOUT_CODE =
      DL-CURR_WORKOUT_CODE
  END IF
  DL-WOUT-START-DATE-6 = CURRENT SYSTEM DATE

/* ONLY UPDATE PRO/RE WHEN NOT = X FOR NEVER RESOLICIT
  IF DL-WORKOUT-INIT-CODE NOT = 'X' THEN
    WORKOUT-INIT-CODE = 'R'
  END IF

/*UPDATE STATUS CHANGE DATE
  DL-WOUT-STAT-DATE-6 = CURRENT SYSTEM DATE
/*CURRENT STATUS TO PENDING
  DL-WORKOUT-STAT-CODE = 'P'
/* TRANSLATE FAST TRACK WORKOUT TYPE TO REQ CODE

```

```

IF CURR-WORKOUT-CODE > ' ' THEN
    PREV-WORKOUT-CODE = CURR-WORKOUT-CODE
END IF
IF WORKOUT_TYPE = 'REPAY/FORBEAR'
    DL-CURR-WORKOUT-CODE = 'F'
END IF
IF WORKOUT_TYPE = 'MODIFICATION'
    DL-CURR-WORKOUT-CODE = 'M'
END IF
IF WORKOUT_TYPE = 'BAP'
    DL-CURR-WORKOUT-CODE = 'B'
END IF
IF WORKOUT_TYPE = 'BAPMOD'
    DL-CURR-WORKOUT-CODE = 'BM'
END IF
IF WORKOUT_TYPE = 'PRESALE'
    DL-CURR-WORKOUT-CODE = 'P'
END IF
IF WORKOUT_TYPE = 'DEED-IN-LIEU'
    DL-CURR-WORKOUT-CODE = 'D'
END IF
/* SET EVAL STATUS TO APPROVED
    DL-CURR-WORKOUT-IND = 'A'
/* BY EFT = E-LMO FAST TRACK
    DL-CURR-WORKOUT-BY = 'EFT'
/*
    DL-CURR-WORKOUT-DATE-6 = CURRENT SYSTEM DATE
/* HANDLE CENTURY SETTING APPROPRIATELY

/* SET FILE LOC TO THE INITIALS OF THE WO REP ASSIGNED TO THE APPROVED WO
    DL-FILE-LOC = 'F-' + WO_REP_INITIALS
/* SET FOLLOWUP CODE TO WORKOUT IN PROGRESS
    DL-FOLLOW-UP-CODE = 'WP'
/* FOLLOW UP DATE IS SET BY BUSINESS LOGIC IN E-LMO FAST TRACK
/* SEE SPECSAPPROVAL.DOC FOR MORE INFORMATION
    DL-FOLLOW-UP-DATE-6 = FU_DATE
/* HANDLE CENTURY SETTING APPROPRIATELY
/* UPDATE FORECLOSURE START, SALE DATES ONLY IF REO DOES NOT HAVE FC
STARTED /* AND BORROWER HAS NOW DECLARED BK
    IF DL-START-DATE-6 OR DL-SALE-DATE-6 HAS A VALID DATE IN IT
/* ANY CONDITIONS ON THE DATES BEING IN CURRENT FISCAL YR??
/* HANDLE CENTURY SETTING
    IF BANKRUPTCY_STATUS NOT = 'CHP 7 ACTIVE '
        START-DATE-6 = FC_START_DATE /*PASSED IN ON TRANS
        SALE-DATE-6 = FC_SALE_DATE /*PASSED IN ON TRANS
    END IF
END IF

/*CHANGE CAT CD ON AN APPROVED PRESALE
/* WHAT ABOUT DIL????
    IF WORKOUT_TYPE = 'PRESALE'
        DL-MG-DELQ-CATEGORY = '40'

/*UPDATE LAST UPDATE DATE
    DL-LST-DELQ-UPDT-DATE-6 = CURRENT SYSTEM DATE

```


END-EXEC.

Else

/* REO_INVEST_WORKOUT shows a different servicer than servicer name assigned to user

/* get lendr-name-key for the user assigned servicer name

EXEC SQL

SELECT LENDR_NAME_KEY FROM REO_LENDR_NAME

WHERE LENDR_NAME = servicer_name from transaction

AND INVESTOR CODE = investor_name passed in on transaction

END-EXEC.

/* if exists for this investor then update reo_invest_workout setting investor_code,

/*lendr_name_key, workout_pgm_ind, AND user_initials

IF SQLCODE EQUAL ZERO

EXEC SQL

UPDATE REO_INVEST_WORKOUT SET CERT_NBR = cert_nbr,

INVESTOR_CODE = investor_name, USER_INITIALS = 'F-EFT',

WORKOUT_PGM_IND = workout_pgm_ind, INVESTOR_LOAN_NBR =

investor_loan_nbr, LENDER_LOAN_NBR = lender_loan_nbr,

LENDR_NAME_KEY = REO_LENDR_NAME.LENDR_NAME.KEY from
above select

END-EXEC.

ELSE

INSERT INTO REO_LENDR_NAME VALUES (servicer_name from

transaction, investor_code from transaction, workout_pgm_ind from

transaction, ' ', ' ', 'F-EFT')

NOTE – need to get LENDR_NAME_KEY from above insert

INSERT INTO REO_INVEST_WORKOUT VALUES (cert_nbr,

investor_name, ' ', 'F-EFT', workout_pgm_ind, ' ', ' ', ' ', ' ',

investor_loan_nbr, lender_loan_nbr, ' ', ' ', NULL, ' ', ' ', NULL,

LENDR_NAME_KEY from above insert, NULL, ' ', ' ', ' ', ' ', 0, ' ')

Error with Error-Code = 703

Error-Field = REO_LENDR_NAME.LENDR_NAME

Error-Data = REO_LENDR_NAME.INVESTOR_CODE

LMO err_codes table message_txt for err_code_txt = 703 will be

Message Text ' New Servicer/Investor entry added to Lender Name Table.

Notify Joyce McKenzie to set AMBASSASOR_IND appropriately.'

End If

End If

Else

/* no investor record exists in DB2 for this certificate – set one up

EXEC SQL

SELECT LENDR_NAME_KEY FROM REO_LENDR_NAME

WHERE LENDR_NAME = servicer_name from transaction

AND INVESTOR CODE = investor_name passed in on transaction

END-EXEC.

IF SQLCODE EQUAL ZERO

EXEC SQL

09766133 "021601

```

INSERT INTO REO_INVEST_WORKOUT VALUES (cert_nbr, investor_name, 'F-EFT', workout_pgm_ind, investor_loan_nbr, lender_loan_nbr, LENDR_NAME_KEY, 0)
END-EXEC.

```

ELSE

```

INSERT INTO REO_LENDR_NAME VALUES (servicer_name from
transaction, investor_code from transaction, workout_pgm_ind from
transaction, 'F-EFT')

```

NOTE – need to get LENDR_NAME_KEY from above insert

```

INSERT INTO REO_INVEST_WORKOUT VALUES (cert_nbr,
investor_name, 'F-EFT', workout_pgm_ind,
investor_loan_nbr, lender_loan_nbr, NULL, NULL,
LENDR_NAME_KEY from above insert, NULL, 0)

```

```

Error with Error-Code = 703
Error-Field = REO_LENDR_NAME.LENDR_NAME
Error-Data = REO_LENDR_NAME.INVESTOR_CODE

```

LMO err_codes table message_txt for err_code_txt = 703 will be
 Message Text ' New Servicer/Investor entry added to Lender Name Table.
 Notify Joyce McKenzie to set AMBASSASOR_IND appropriately.'

End If

End if

P-DELQ-REC-SETUP. (not yet complete)

```

INSERT G8-CERT-REC
CERT_NBR,
INSERT G8-DELQ-REC
CAT_CD ??, LOAN DUE DATE, CURR UPB
??? ANY OTHER RECS REQUIRED

```

P-IDMS-FAILED-UPDATE.

IF G8-CERT-REC NOT FOUND FOR CERT_NBR+ Master Policy Nbr ION

```

Create error record with
Error-Code = 107
Error-Field = '
Error-Data = '
Message txt for error code 107 'Record not found for cert'.

```

Exit Processing.

END IF

If (G8-CERT-REC first 2 positions of FILE-LOC = 'C#' or 'C-') or (CM-FILE-LOC not = 'F-EFT')
 (then certificate is in Claims)

```

Error with Error-Code = 122
Error-Field = FILE-LOC
Error-Data = '

```

LMO err_codes table message_txt for err_code_txt = 122 will be
 Message Text 'Current workout cannot be marked as Failed. Please contact GE for assistance -
 see Contact Us.'

Exit Processing.

End If

IF G8-DELQ-REC not found for cert then

Error with Error-Code = 108

Error-Field = 'G8-DELQ-REC'

Error-Data = ''

LMO err_codes table message_txt for err_code_txt = 108 will be

Message Text 'Physical Database Error Occurred - Please Contact GE for assistance.'

Exit processing.

End If.

/*UPDATE STATUS CHANGE DATE

DL-WOUT-STAT-DATE-6 = CURRENT SYSTEM DATE from transaction

/*CURRENT STATUS TO PENDING

DL-WORKOUT-STAT-CODE = 'N'

/* SET EVAL STATUS TO APPROVED

DL-CURR-WORKOUT-IND = 'F'

/*UPDATE LAST UPDATE DATE

DL-LST-DELQ-UPDT-DATE-6 = CURRENT SYSTEM DATE

P-IDMS-NOTEPAD-INSERT. (not completed yet)

Move the text passed in on the transaction to working storage which is redefined as

Pic X(55) occurs 50 times.

OBTAIN CALC on G8-DELQ-REC for this cert-nbr key to be sure you are current on the correct
 IDMS record.

/* Insert Notepad Header Record

/* Insert Notepad Text Record

DATE - CURRENT DATE

BY = 'EFT'

TEXT = text passed in on transaction for each 55-byte length that is nonblank.

Application Functions

Description: Business logic used to determine Approval on a workout. The first function ValidWORequest runs based on the data entered in the web application and for the requested workout type. The code runs a series of edits and either declares the request valid or errors it. If it errors, the user is presented with the Does Not Meet Guidelines screen displaying the respective message. The second function SetApprovalConditions contains the logic required to build the Approval web page to show the user the terms and conditions of the approval.

Function ValidWORequest (requestedWO As String, internalIDNbr As String)

*** this function will return either

*** a) FALSE, sErrCode, sErrField, sErrData, a proceed message, and up to 3

*** proceed options OR

*** b) TRUE

'Workout Guidelines – Feasibility Edits

ValidWORequest = True ' if no errors

*** to handle edit on REO data for Valid WO

select * from loans where loans.internal_id_nbr = internalIDNbr

If oLoans.workout_cd = 'SYSBLK'

 If requestedWO in ('BAP', 'BAPMOD') then

 'prior BAP Edit

 If oLoans.previous_BAP_adv_Amt > 0 then

 'error message = 'Unpaid Balance Exists on a prior BAP.'

 SErrCode = '506'

 proceedMsgBegin1

 in a table then show

 proceedOption1

 proceedOption2

 proceedOption3

 ProceedOption6

 ValidWORequest = False

 End If

 Else

 Return

 End If

End if

select * from borrowers where borrowers.internal_id_nbr = internalIDNbr and borr_cd = 'PRI'

select * from subj_property where subj_property.internal_id_nbr = internalIDNbr

'define Does not meet guidelines options on how to proceed

proceedMsgBegin1 = 'You can : '

proceedMsgBegin2 = 'Please '

proceedMsgBegin3 = 'You can: Increase contribution from borrowers OR'

proceedMsgBegin4 = 'You can: Increase sales price OR reduce closing costs OR'

proceedMsgBegin5 = 'You can : Increase sales price OR reduce closing costs, Increase Borrower

Contribution of Cash or Cash Equivalent P-Note OR '

msg6 and msg7 have been deleted

'the click here link will close this window and return user to MainPage

proceedOption1 = 'Choose another workout. Click here.'


```

'the click here link will close this window and link to www.e-lmo.com
If security_level = 93 then
'user is not authorized to do full data entry into LMO
    proceedOption2 = 'Choose the full data entry option. Click here.'
Else
    proceedOption2 is not allowed as an option so do not include
End If

'the click here link will close this window and load the Submit Package form
proceedOption3 = 'Submit a completed package to GE for review. Click here.'
proceedOption4 = 'Choose to do a MOD, Presale or Deed-in-lieu workout for this loan.'
proceedOption5 = 'Choose a Presale or Deed-in-lieu workout for this loan.'
'click here will take the user to the Contact Us screen.
'NOTE – this proceed option will be added to ALL cases where validWORequest is False
proceedOption6 = 'Contact GE for assistance. Click Here.'
NOTE
    SErrField = Blank for ALL messages UNLESS OTHERWISE NOTED
    SerrData = Blank for ALL messages UNLESS OTHERWISE NOTED
****
'setting bankruptcyStatus
dim bankruptcyStatus as string

Select chp7_ind, chp7_des, other_bankruptcy_des from borrowers
where borrowers.internal_id_nbr = internalIDNbr (as passed into function) and borr_cd = 'PRI'

If oBorrowers.chp7_ind = 'Y' then
    If oBorrowers.servicer_bkrpt_status_ind = 'R' then
        bankruptcyStatus = 'Chp 7 Reaffirmed Discharged'
    else
        If oBorrowers.servicer_bkrpt_status_ind = 'N' then
            bankruptcyStatus = 'Chp 7 NonReaffirmed Discharged'
        else
            if oBorrowers.servicer_bkrpt_status_ind = '' then
                bankruptcyStatus = 'Chp 7 Active'
            endif
        End If
    End If
else
    If oBorrowers.Chp7_ind = 'N' or '' then
        if oBorrowers.other_bankruptcy_des = 'Chapter 11' then
            bankruptcyStatus = 'Chapter 11 Active'
        else
            if oBorrowers.other_bankruptcy_des = 'Chapter 13' then
                bankruptcyStatus = 'Chapter 13 Active'
            else
                bankruptcyStatus = 'N/A'
            End if
        END If
    End If
End If

'These edit applies for ALL workouts'
If bankruptcyStatus = 'Chp 7 Active' then
    'no workout options are allowed on active chp 7 bankruptcies

```

```

'fatal error for this workout
  sErrCodeValue = '501'
  proceedMsgBegin1
  proceedOption3
  proceedOption6
  ValidWORequest = False

```

```
End if
```

```

estNbrDaysDelinquent = nbr days difference between current system date and
                      oLoans.loan_due_date

```

```

If (oLoans.category_cd= 'ND' or '00') and estNbrDaysDelinquent > 30 then
  'fatal error for this workout'
  sErrCodeValue = '603'
  ValidWORequest = False

```

```

Note -err message = 'GE does not show this loan as delinquent. Please contact GE
Delinquency Reportg at 1-800-334-9270')

```

```
End if
```

```

'edit on Investor Name selected
If oLoans.investor_Name= 'Freddie Mac' then
  SErrCode = '502'
  ValidWORequest = False

```

```

Error Message 'Cannot submit a workout for this loan. Loan is part of delegated
program.'

```

```
End if
```

```
****
```

```

Calculate daysDiff = #days difference between (oLoans.orig_date and oLoans.Loan_due_date)
Calculate mosDiff = # months difference between (oLoans.orig_date and oLoans.Loan_due_date)

```

```
If (daysDiff <= 30) or (mosDiff <=12) then
```

```

  'a warning message should be logged to indicate
  'rep will be responsible for notifying Investigations upon approval via Workflow Email
  Insert into message_info values (@sysDateTime, @siteID, @certNbr,
  'MAX_VALUES_FT_', '701', oLoans.orig_date, oLoans.LoanDueDate)
  'Just a warning so validWORequest should still = TRUE

```

```
End if
```

```
*****
```

```
'This edit applies for ALL CURE related workout requests
```

```
If requestedWO = 'Repay/Forbear' then
```

```

  If oLoans.nbr_months_delq > 24 months
    SErrCode = '503'
    ProceedMsgBegin2
    ProceedOption3
    ProceedOption5
    ProceedOption6
    ValidWORequest = False

```

```
'error message = 'Loan is greater than 24 months delinquent.'
```

```
End If
```

```
End If
```

```
'if passes above edit then continue with the following select stmt
```

C:\Program Files\Microsoft SQL Server\90\Tools\Binn\sqlcmd.exe

Select Case reqWorkoutType

Case "Repay/Forbear"

'Foreclosure Start Edit/Sale Date EDIT REMOVED -> replaced with a Condition on the Approval letter that a Stipulated Agreement must be returned to GE.

Select * from repay where repay.internal_id_nbr = internalIDNbr

'repayment term edit

' max_values.ft_RepayTerm = 18 mos.

If repay.term_w_curr_disp_amt > max_values.ft_Max_Repay_Term then

SErrCode = '504'

proceedMsgBegin1

in a table then show

proceedOption1

proceedOption2

proceedOption3

ProceedOption6

ValidWOREquest = False.

'error message = 'Repay Term exceeds maximum allowable term of ' &
max_values.ft_RepayTerm & ' months.'

End If

'Bankruptcy edit

' If bankruptcy status is Chp 7 Reaffirmed or Chp13 then conditional approval will handle the

' request for documentation (e.g. Reaffirmation agreement) to prove borrower willingness

' to enter this arrangement and/or that this is not an active bankruptcy'

'NOTE – language on conditional approval will dictate what documentation is required

If bankruptcyStatus = 'Chp 7 NonReaffirmed Discharged' OR 'Chapter 11' then

if oLoans.est_fclsr_date = '01/01/1900' or oLoans.fclsr_sale_date = '01/01/1900' then

'loan has NOT been referred to Foreclosure'

SErrCode = '505'

proceedMsgBegin2

proceedOption3

proceedOption4

ProceedOption6

ValidWOREquest = False.

'error message = 'Loan indicates bankruptcy. '

End If

End If

ValidWOREquest = True

Case "BAP"

Select * from bap where bap.internal_id_nbr = internalIDNbr

'prior BAP Edit

If oLoans.previous_BAP_adv_Amt > 0 then

'error message = 'Unpaid Balance Exists on a prior BAP.'

SErrCode = '506'

proceedMsgBegin1

in a table then show

proceedOption1

proceedOption2

```

        proceedOption3
        ProceedOption6
        ValidWORequest = False
    End If

```

```

'category code edit
If month portion of oLoans.loan_Due_Date > current month then
    'error message = 'Loan is not delinquent. '
    SErrCode = '507'
    proceedMsgBegin1
    in a table then show
    proceedOption1
    proceedOption2
    proceedOption3
    ProceedOption6
    ValidWORequest = False
End If

```

'Occupancy Edit
 '??? Check if REO has property use info, if so then if Rental or Residential and Occupancy is
 Tenant then pass the occupancy edit.

```

    If subj_property.occupancy_des is NOT 'Owner' or 'Owner & Tenant' or 'Vacant - Occupied
W/in 90 Days' then
        'error message = 'Property must be occupied. '
        sErrCode = '508'
        proceedMsgBegin1
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6

        ValidWORequest = False.
    End If

```

```

'Bankruptcy Edit – if other than Chp7 Reaffirmed then consider as Active Bankruptcy
    If bankruptcyStatus = 'Chp 7 NonReaffirmed Discharged' OR 'Chapter 11' then
        SErrCode = ' 505'
        proceedMsgBegin1
        in a table then show
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6

    End If

```

```

'Property value / date edit
'max_values.ft_NbrBPODays = 180
    If oSubj_property.date_of_value_date <= (current Date – max_values.ft_NbrBPODays) then
        'error message = 'Property Valuation Date is older than 90 days. Please obtain a recent
        'valuation. '
        sErrCode = '509'
        ValidWORequest = False
    End If

```

```

If oSubj_property.as_is_value_amt is ??? determine condition for tolerance
    'error message = ?????'
    ValidWOREquest = False
End If

```

```

    'Equity Edit
    If (oLoans.curr_upb_amt * oSubj_property.severity_amt) < (max_values.ft_F MVPct *
oSubj_property.as_is_value_amt) then
        'error message = 'Does not meet guidelines. '
        sErrCode = '519'
        sErrField = 'Property Equity'
        proceedMsgBegin1
        in a table then show
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6
        ValidWOREquest = False

```

```

End IF

```

```

    'calc Max BAP Advance amount for BAP or BAPMODs WITHOUT Capitalization Amt
    ' - see below for max bap for BAP MOD with Cap
    If oLoans.workout_cd = 'BAP' or 'BAPMOD Rate' or 'BAPMOD Term' or
        'BAPMOD RateTerm' then
        maxBAPAmtReqd = (reinstateAmt - priBorrContribAmt)
    Else
        maxBAPAmtReqd = (reinstateAmt - priBorrContribAmt - capitalizationAmt entered)
        If maxBAPAmtReqd <= 0 then
            'error message = 'BAP is not required to bring loan current given the capitalization
            ' amount.
            SErrCode = '511'
            proceedMsgBegin1
            in a table then show
            proceedOption1
            proceedOption2
            proceedOption3
            ProceedOption6
            ValidWOREquest = False

```

```

End If

```

```

    'max amount to be advanced is the lesser of 50% of exposure, $15K or amount needed to bring
    'current.

```

```

        maxBAPViaExposure = ( max_values.ft_BAPExposurePct * (oLoans.curr_upb_amt *
oSubj_property.severity_amt *
        coveragePct)
        If maxBAPAmtReqd > maxBAPViaExposure then
            If maxBAPViaExposure > max_values.ft_BAPLimitAmt then
                maxBAPAllowedAmt = max_values.ft_BAPLimitAmt
                addtlContrib = maxBAPAmtReqd - maxBAPAllowedAmt
                'error message = "BAP Amount Required Exceeds Guidelines. Additional
                Borrower Contribution is required for approval.'
                SErrCode = '512'
                SErrField = 'MaxBAP $' & maxBAPAllowedAmt
                SDataField = 'Addtl Contrib $' & addtlContrib

```

TOTAL 201001

```

ValidWORequest = False
Else
    maxBAPAllowedAmt = maxBAPViaExposure
    addtlContrib = maxBAPAmtReqd - maxBAPAllowedAmt
    'error message = "BAP Amount Required Exceeds Guidelines. Additional
    Borrower Contribution is required for approval.'
    SErrCode = '512'
    SErrField = 'MaxBAP $' & maxBAPAllowedAmt
    SDataField = 'Addtl Contrib $' & addtlContrib
    ValidWORequest = False
End If
Else
    If maxBAPAmtReqd > max_values.ft_BAPLimitAmt then
        maxBAPAllowedAmt = max_values.ft_BAPLimitAmt
        addtlContrib = maxBAPAmtReqd - maxBAPAllowedAmt
        'error message = 'BAP Amount Required Exceeds Guidelines. Additional
        Borrower Contribution is required for approval.'
        SErrCode = '512'
        SErrField = 'MaxBAP $' & maxBAPAllowedAmt
        SDataField = 'Addtl Contrib $' & addtlContrib
        ValidWORequest = False
    Else
        'otherwise OK with maxBAPAmtReqd is OK without addtlContrib and so workout is OK
        maxBAPAllowedAmt = maxBAPAmtReqd
        addtlContrib = 0
    End If
End If

'Affordability Edit
If (oBAP.dispos_inc_used_amt > 0) then
    'then determine possible payment options which would leave
    oBAP.dispos_inc_used_amt positive
    'start with 10 years (120mos) and move back by 6 months each time until disposable
    'income goes negative. Use the MinNoteTerm to prefill the BAP Note term/payment
    'dropbox, incrementing by 6 months up to 120 months.
    nbrMos = 120
    Do While (newDisposableIncome > 0)
        newDisposableIncome = (oBAP.dispos_inc_used_amt - (maxBAPAllowedAmt /
nbrMos))
        If newDisposableIncome > 0 then
            MinNoteTerm = nbrMos
            nbrMos = nbrMos - 6
        end if
    Loop
    'reset min note term since this last iteration caused oBAP.dispos_inc_used_amt to go
negative
    MinNoteTerm = nbrMos + 6
    'check affordability - if borrower can't make note payment at 120 months then fail BAP
    If (MinNoteTerm = 120) And
        ( (oBAP.dispos_inc_used_amt - (maxBAPAllowedAmt / nbrMos)) < 0) then
        'error message = 'Borrower can not afford minimum Note Payment.'
        SErrCode = '513'
        proceedMsgBegin1
        in a table then show
        proceedOption1

```

```

        proceedOption2
        proceedOption3
        ProceedOption6
        ValidWORequest = False.

```

```

    End If

```

```

    Case "BAPMOD"

```

```

        bapmodTypeCd = Upper(substr(oLoans.workout_cd, 7, 40))
        Select * from bap_payment where bap_payment.internal_id_nbr = internalIDNbr and
        bapmod_type_cd = rtrim(bapmodTypeCd)

```

```

        calculate maxCAPGEInsure = the lesser of
        capUsingOrigBal = ( oLoans.orig_UPB_amt * (select maxOverUPB from workout.max_values) ) -
        oLoans.current_upb_amt )
        OR
        capUsingValue = (oLoans.orig_LTV_amt / 100) * oSubj_property.as_is_value_amt

```

```

        'prior BAP Edit
        If oLoans.previous_BAP_adv_Amt > 0 then
            'error message = 'Unpaid Balance Exists on a prior BAP. '
            sErrCode = '514'
            proceedMsgBegin1
            in a table then show
            proceedOption1
            proceedOption2
            proceedOption3
            ProceedOption6
            ValidWORequest = False

```

```

        End IF

```

```

        'category code edit
        If month portion of oLoans.loan_Due_Date > current month then
            errReason = 'Loan is not delinquent.'
            SErrCode = '507'
            proceedMsgBegin1
            in a table then show
            proceedOption1
            proceedOption2
            proceedOption3
            ProceedOption6
            ValidWORequest = False

```

```

        End If

```

```

        'Occupancy Edit

```

```

        '??? Check if REO has property use info, if so then if Rental or Residential and Occupancy is
        Tenant then pass the occupancy edit.

```

```

        If oSubj_property.occupancy_des is NOT 'Owner' or 'Owner & Tenant' or 'Vacant -
        Occupied W/in 90 Days' then
            'error message = 'Property must be occupied. '
            sErrCode = '508'
            proceedMsgBegin1
            in a table then show

```

T E C H N I C A L S P E C I F I C A T I O N

```

        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6
        ValidWORequest = False.
    End If
    'Bankruptcy Edit – if other than Chp7 Reaffirmed then consider as Active Bankruptcy
    If bankruptcyStatus = 'Chp 7 NonReaffirmed Discharged' OR 'Chapter 11' then
        'error message = 'Bankruptcy Type. '
        sErrCode = '505'
        proceedMsgBegin1
        in a table then show
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6
        ValidWORequest = False.
    End If

    'Property value / date edit
    'max_values.ft_NbrBPODays = 180
    If oSubj_property.date_of_value_date <= (current Date – max_values.ft_NbrBPODays) then
        'error message = 'Property Valuation Date is older than 90 days. Please obtain a recent
valuation. '
        sErrCode = '509'
        ValidWORequest = False
    End If

    If oSubj_property.as_is_value_amt is ??? determine condition for tolerance
        'error message = ?????
        ValidWORequest = False
    End If

    If oLoans.new_mod_term > 360 then
        'error message – Term exceeds guidelines for maximum term of 360 months.'
        sErrCode = '520'
        sErrField = 'Term'
        sErrData = oLoans.new_mod_term
        ProceedMsgBegin1
        ProceedOption1
        ProceedOption2
        ProceedOption3
        ProceedOption6
        ValidWORequest = False
    End if

    'Equity Edit
    If (oLoans.curr_upb_amt * oSubj_property.severity_amt) > (max_values.ft_FMVPct *
oSubj_property.as_is_value_amt) then
        'error message = 'Does not meet Guidelines'
        sErrCode = '510'
        sErrField = 'Property Equity'
        proceedMsgBegin1
        in a table then show
        proceedOption1

```



```

    proceedOption2
    proceedOption3
    ProceedOption6
    ValidWORequest = False

```

```

End IF

```

```

'calc Max BAP Advance amount for BAP or BAPMODs WITHOUT Capitalization Amt
' - see below for max bap for BAP MOD with Cap

```

```

If oLoans.workout_cd = 'BAP' or 'BAPMOD Rate' or 'BAPMOD Term' or
'BAPMOD RateTerm' then
    maxBAPAmtReqd = (reinstateAmt - priBorrContribAmt)

```

```

Else
    maxBAPAmtReqd = (reinstateAmt - priBorrContribAmt - capitalizationAmt entered)
    If maxBAPAmtReqd <= 0 then
        'error message = 'BAP is not required to bring loan current given the capitalization
        ' amount.
        SErrCode = '511'
        proceedMsgBegin1
        in a table then show
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6
        ValidWORequest = False
    
```

```

End If

```

```

'max amount to be advanced is the lesser of 50% of exposure, $15K or amount needed to bring
'current.

```

```

    maxBAPViaExposure = ( max_values.ft_BAPExposurePct * (oLoans.curr_upb_amt
    *oSubj_property.severity_amt *
    coveragePct)

```

```

    If maxBAPAmtReqd > maxBAPViaExposure then

```

```

        If maxBAPViaExposure > max_values.ft_BAPLimitAmt then
            maxBAPAllowedAmt = max_values.ft_BAPLimitAmt
            addtlContrib = maxBAPAmtReqd - maxBAPAllowedAmt
            'error message = 'BAP Amount Required Exceeds Guidelines. Additional
            Borrower Contribution is required for approval.'
            SErrCode = '512'
            SErrField = 'MaxBAP $' & maxBAPAllowedAmt
            SDataField = 'Addtl Contrib $' & addtlContrib
            ValidWORequest = False
        
```

```

    Else

```

```

        maxBAPAllowedAmt = maxBAPViaExposure
        addtlContrib = maxBAPAmtReqd - maxBAPAllowedAmt
        'error message = 'BAP Amount Required Exceeds Guidelines. Additional
        Borrower Contribution is required for approval.'
        SErrCode = '512'
        SErrField = 'MaxBAP $' & maxBAPAllowedAmt
        SDataField = 'Addtl Contrib $' & addtlContrib
        ValidWORequest = False
    
```

```

    End If

```

```

Else

```

```

    If maxBAPAmtReqd > max_values.ft_BAPLimitAmt then
        maxBAPAllowedAmt = max_values.ft_BAPLimitAmt
    
```

```

        addtlContrib = maxBAPAmtReqd - maxBAPAllowedAmt
        'error message = 'BAP Amount Required Exceeds Guidelines. Additional
        Borrower Contribution is required for approval.'
        SErrCode = '512'
        SErrField = 'MaxBAP $' & maxBAPAllowedAmt
        SDataField = 'Addtl Contrib $' & addtlContrib
        ValidWORequest = False
        'otherwise OK with maxBAPAmtReqd is OK without addtlContrib and so workout is OK
    End If

    'Affordability Edit
    If (oBAP.dispos_inc_used_amt > 0) then
        'then determine possible payment options which would leave
    oBAP.dispos_inc_used_amt positive
        'start with 10 years (120mos) and move back by 6 months each time until disposable
        'income goes negative. Use the MinNoteTerm to prefill the BAP Note term/payment
        'dropbox, incrementing by 6 months up to 120 months.
        nbrMos = 120
        Do While (newDisposableIncome > 0)
            newDisposableIncome = (oBAP.dispos_inc_used_amt - (maxBAPAllowedAmt /
nbrMos))

            If newDisposableIncome > 0 then
                MinNoteTerm = nbrMos
                nbrMos = nbrMos - 6
            end if

        Loop
        'reset min note term since this last iteration caused oBAP.dispos_inc_used_amt to go
        negative
        MinNoteTerm = nbrMos + 6
        'check affordability - if borrower can't make note payment at 120 months then fail BAP
        If (MinNoteTerm = 120) And
            ((oBAP.dispos_inc_used_amt - (maxBAPAllowedAmt / nbrMos)) < 0) then
            'error message = 'Borrower can not afford minimum Note Payment.'
            SErrCode = '513'
            proceedMsgBegin1
            in a table then show
            proceedOption1
            proceedOption2
            proceedOption3
            ProceedOption6
            ValidWORequest = False.
        End If

    Case "MOD"

        modTypeCd = Upper(substr(oLoans.workout_cd, 4, 40))
        Select * from mod where mod.internal_id_nbr = internalIDNbr and mod_type_cd =
rtrim(modTypeCd)

        'bankruptcy edit - no code edits. Approval conditions will dictate documentation required if
        bankruptcy status is other than Chapter 11 or Chp 7 Reaffirmed.
        'Can accept MODs on active bankruptcies from Fannie/Freddie with Investor Rider
        'Condition of approval is receipt of Investor Rider Document .
        'Can accept MODS on active bankruptcies for private investor as long as receive
        documentation with same language as is in Fannie/Freddie Rider.

```

'Note – approval letter will show
 'Capitalization Amount : amt user entered for cap
 'Modified Loan Balance = curr UPB + entered CAP
 'GEInsured Amount = up to 105% of original UPB

If oLoans.new_mod_term > 360 then
 'error message – Term exceeds guidelines for maximum term of 360 months.'
 sErrCode = '520'
 sErrField = 'Term'
 sErrData = oLoans.new_mod_term
 ProceedMsgBegin1
 ProceedOption1
 ProceedOption2
 ProceedOption3
 ProceedOption6
 ValidWORequest = False
 End if

Case " Presale"

Select * from presale where presale.internal_id_nbr = internalIDNbr

If oPresale.presale_closing_date > oLoans.payoff_date then
 'error message = 'Presale Closing Date must be less than or equal to Payoff Date.'
 sErrCode = '521'
 sErrField = 'Presale Closing Date'
 sErrData = oPresale.presale_closing_date
 validWORequest = False
 End If

If oPresale.presale_closing_date > current system date + 60 days then
 'error message = 'Presale must close within 60 days from current date.'
 sErrCode = '522'
 sErrField = 'Presale Closing Date'
 sErrData = oPresale.presale_closing_date
 validWORequest = False

'if loan is current, then use the following edits for feasibility

If oLoans.category_Cd = 'ND' or '00' then

 netProceeds = osubj_property.offer_amt – oPresale.closing_Cost_amt
 If netProceeds + oBorrowers.initial_cntrb_amt where borr_cd = 'PRI') <
 oLoans.actual_payoff_Amt then
 'error message = 'Net Proceeds + contribution does not meet full loss. Additional
 contribution is required.'
 SErrCode = '515'
 SErrField = 'Addtl Contribution'
 SErrData = (oLoans.actual_payoff_Amt – netProceeds)
 ProceedMsgBegin5
 in a table then show
 proceedOption1
 proceedOption2
 proceedOption3
 ProceedOption6

Else

```

'loan is delinquent, so perform the following edits
'bankruptcy edit
If bankruptcyStatus = "Chp 7 NonReaffirmed Discharged" OR 'Chp 11' OR 'Chp 13' then
  If (oSubj_property.date_of_value_date <= (current Date -
max_values.ft_NbrBPODays)) then
    'error message = 'Property Valuation Date is older than 90 days. Please obtain
    a 'recent valuation. '
    sErrCode = '509'
    ValidWORequest = False

```

End If

```

If (oSubj_property.as_is_value_amt) > 0 And ( ??? determine condition for tolerance)
  'error message = 'Required minimum Net On Sale not met. '
  sErrCode = '516'
  proceedMsgBegin4
  in a table then show
  proceedOption1
  proceedOption2
  proceedOption3
  ProceedOption6

```

ValidWORequest = False

End If

```

If (oSubj_property.actual_net_on_sale_amt < workout.max_values.nosPct *
oSubj_property.as_is_value_amt ) then
  'error message = 'Required minimum Net On Sale not met. '
  sErrCode = '516'
  proceedMsgBegin4
  in a table then show
  proceedOption1
  proceedOption2
  proceedOption3
  ProceedOption6

```

ValidWORequest = False

'Fair Market Value edit – check that NOS is at least x% of value

```

  If (oSubj_property.as_is_value_amt- oLoans.actual_payoff_Amt) <
(oSubj_property.as_is_value_amt* workout.max_values.nosPct) then
    'error message = 'Required Net On Sale not met.'
    sErrCode = '516'
    proceedMsgBegin4
    in a table then show
    proceedOption1
    proceedOption2
    proceedOption3
    ProceedOption6
    ValidWORequest = False

```

End If

'Loss Amt Edit

```

  estLossAmt = oLoans.actual_payoff_Amt - (oSubj_property.offer_amt -
oPresale.closing_Cost_amt)
  If estLossAmt > max_values.ft_LossLimitAmt then
    'error message = 'Maximum Loss Limit Exceeded. '

```

FOR THE "OFFER" OF THE

```

        sErrCode = '517'
        proceedMsgBegin4
        in a table then show
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6
        ValidWORequest = False
    End If

Else
    If bankruptcy status 'Chp7 Reaffirmed Discharged' or 'N/A'
        'all other edits pertain to ANY OTHER TYPE of Bankruptcy??
        'payoff to closing date edit. Payoff Date must be = Closing Date included at screen level
        'Alert message = 'Payoff Date must equal Closing Date.'

        'Property value / date edit
        If oSubj_property.date_of_value_date <= (current Date - max_values.ft_NbrBPODays)
then
            'error message = 'Property Valuation Date is older than 90 days. Please obtain
            a 'recent valuation. '
            sErrCode = '509'
            ValidWORequest = False
        End If
        If (oSubj_property.as_is_value_amt) > 0 And ( ??? determine condition for tolerance)
            'error message = ??????
            ValidWORequest = False
        End If

        'required NOS / Net Proceeds edit
        If oSubj_property.as_is_value_amt > 0 And oLoans.actual_payoff_amt > 0 then
            reqdNetOnSale = fastTractNetProceedsPct * oLoans.actual_payoff_amt
            '
            estNetProceedsOfSale = (salesPriceAmt - closingCostAmt + borrCashContrib
            + claimReductionAmt)
            estNetProceedsOfSale = (oSubj_property.offer_amt -
oPresale.closing_Cost_Amt + oBorrowers.initial_contrib_amt where borr_cd = 'PRI'
            + oPresale.act_cntrb_ge_clmreduc_amt)
            if estNetProceedsOfSale < reqdNetOnSale then
                sErrCode = '516'
                proceedMsgBegin5
                in a table then show
                proceedOption1
                proceedOption2
                proceedOption3
                ProceedOption6
                ValidWORequest = False
            End If
        End If

        'Fair Market Value edit - check that NOS is at least x% of value
        If (oSubj_property.as_is_value_amt - oLoans.actual_payoff_amt) <
(oSubj_property.as_is_value_amt * workout.max_values.nosPct) then
            'error message = 'Required Net On Sale not met.'
            sErrCode = '516'
            proceedMsgBegin5

```

```

        in a table then show
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6
        ValidWORequest = False
    End If

    'Loss Amt Edit
    estLossAmt = oLoans.actual_payoff_amt - (oSubj_property.offer_amt -
oPresale.closing_Cost_Amt)
    If estLossAmt > max_values.ft_LossLimitAmt then
        'error message = 'Maximum Loss Limit Exceeded. '
        sErrCode = '517'
        proceedMsgBegin5
        in a table then show
        proceedOption1
        proceedOption2
        proceedOption3
        ProceedOption6

        ValidWORequest = False
    End If
End If

Case "Deed-in-lieu"
Select * from dil where dil.internal_id_nbr = internalIDNbr

    'Property value / date edit
    If oSubj_property.date_of_value_date <= (current Date - max_values.ft_NbrBPODays)
then
        'error message = 'Property Valuation Date is older than 90 days. Please obtain
        a 'recent valuation. '
        sErrCode = '509'
        ValidWORequest = False
    End If
    If (oSubj_property.as_is_value_amt) > 0 And ( ??? determine condition for tolerance)
        'error message = ?????
        ValidWORequest = False
    End If

    'bankruptcy edit
    If bankruptcyStatus = "Chp 7 NonReaffirmed Discharged" then
        expectedClaim = (oLoans.actual_payoff_Amt) - (oSubj_property.as_is_value_amt*
workout.max_values.nos_pct)
        If expectedClaim < (oLoans.Actual_Payoff_Amount * oLoans.coverage_pct) then
            'error message = 'Data does not indicate Loss Over Guaranty.'
            SErrCode = '518'
            proceedMsgBegin1
            in a table then show
            proceedOption1
            proceedOption2
            proceedOption3
            ProceedOption6

```

T E S T E D 0 9 1 6 0 4

```

ValidWORequest = False
Else
    lossOverGuaranteeAmt = expectedClaim - (PayoffAmount * %Covg)
    ValidWORequest = True 'Continue with processing
End If
Else
    'all other edits pertain to ANY OTHER TYPE of Bankruptcy
    'deceased Borrower edit
    '***NOTE - change to a multi select dropdown and fill with borr, cobor and other borr
    'names. User to choose all who are deceased
    Count # of non-blank borrower names (priBorrLastName, coBorrLastName,
        othBorrLastName)
    Count # of entries selected in Deceased Droplist
    If count # borrowers <> count # deceased entries then
        'error message = 'Does not meet guidelines'
        sErrCode = '510'
        proceedMsgBegin1
        proceedOption3
        ProceedOption6
        ValidWORequest = False
    End If
End If

End Select

If ValidWORequest = False then
    Update loans set wo_approved_ind = 'N', wo_complete_date = systemDate
End If

Function SetupDelqInREO
    'call CICS black box with cert#, master policy number,

Function SetApprovalConditions (reqWorkoutType As String)
    'builds HTML for approval conditions required to be displayed on the screen
    'conditions depend on Workout Type and data entered for that workout

Select Case reqWorkoutType

    Case "Repay/Forbear"
        build html line to show 'Conditions of Approval are as follows:'
        build html line to show '1. Loan to be current by ' & value entered in Repay Form for
Expected CureDate & '
        build html line to show '2. Repayment Term is ' & value entered in Repay Form for
Expected Repay Term' & ' months.'

        If loans.est_fclsr_date or loans.fclsr_sale_date > '01/01/1900' then
            'loan is in foreclosure'
            build html line to show '3. Executed Stipulation Agreement must be submitted to
            GE.'

    Case "BAP"
        build html line to show 'The Approval is based on the following information: '
        build html line to show 'Reinstatement Amount: ' & value entered in BAP Form for
ReinstatementAmt

```